AFIT/GST/OS/82M-9

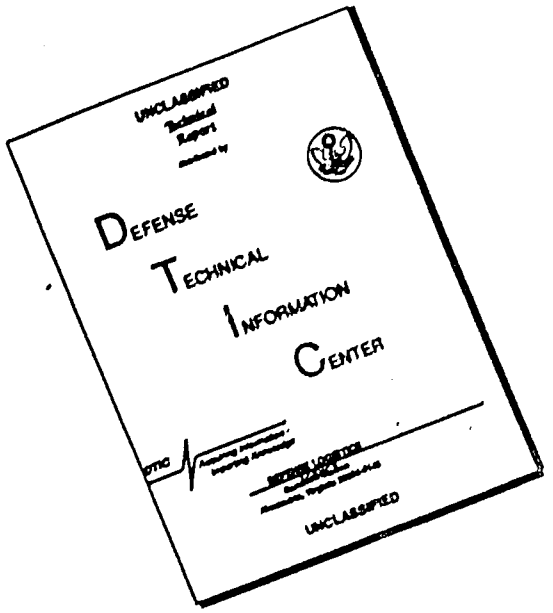
(1)

A USER DEFINABLE SLAM AIRFIELD MODEL DESIGNED FOR EXPERIMENTATION AND ANALYSIS VOLUME II

THESIS

AFIT/GST/OS/82M-9 Robert W. Mann
Maj USAF
Brian J. Shook
2Lt USAF

DISCLAIMER NOTICE



THIS DOCUMENT IS BEST QUALITY AVAILABLE. THE COPY FURNISHED TO DTIC CONTAINED A SIGNIFICANT NUMBER OF PAGES WHICH DO NOT REPRODUCE LEGIBLY.

AFIT/GST/OS/82M-9

Thesis

A USER DEFINABLE SLAM AIRFIELD MODEL DESIGNED FOR *EXPERIMENTATION AND ANALYSIS*

Volume II

by

Robert W. Mann
Maj USAF

and

Brian J. Shook
2Lt USAF

Prepared in partial
fulfillment of
requirements for a
Master's Degree

March 1982

School of Engineering
Air Force Institute of Technology
Wright-Patterson Air Force Base
Ohio

Volume II

Table of Contents

List of Figures

Appendix A: SLAM Coding and Structural Model

This appendix contains the entire SLAM coding required to execute the airfield model on the Control Data Corporation (CDC) 6600 system at Aeronautical Systems Division, Wright-Patterson AFB, Ohio.

The graphical depictions of the structural model are inserted following each functional section of the code listing.

```
1           .*JOB CONTROL LANGUAGE
2          .*   THE MODEL WILL RUN WITH JUST UNDER CM250000.  CP TIME WITHOUT ANY
3          .*TRACE WILL BE UNDER T180.  INPUT/OUTPUT TIME WILL BE UNDER I0360.
4          .*THE LIMITS SHOWN WERE ADEQUATE TO GENERATE 900000 LINES OF OUTPUT
5          .*WITH TRACE FOR SIX SQUADRONS OF SIXTEEN AIRCRAFT WITH REPLACEMENT
6          .*SQUADRONS REQUIRED WHEN OPERATIONAL AIRCRAFT IN A SQUADRON FELL
7          .*BELOW 12.  THE SUPPORTING FORTRAN IS COMPILED EXTERNALLY AND
8          .*STORED IN AN IFS FILE WHICH IS ACCESSED BY THE JCL.
9          .*
10         RWM,CM250000,T240,I0400.  T800845,MANN,BOX4566,AFIT,AFIT,AFIT,91,91,91
11         ATTACH,PROCFIL,ID=A810171,SN=ASDAD.
12         BEGIN,NOSFILE.
13         GET,FZZCBIN,ID=COVEY.
14         REWIND,FZZCBIN.
15         ATTACH,PROCFIL,SLAMPROC,ID=AFIT.
16         BEGIN,SLAM,,M=FZZCBIN,PL=100000.
```

```
17          ; SIMULATION CONTROL STATEMENTS
18          ;    THE GEN CARD IS STANDARD.  THE LIMITS CARD ALLOWS FOR 99 FILES.
19          ; FILE 46 IS NOT CURRENTLY USED.  48 ATTRIBUTES ARE USED.
20          ; ATTRIBUTES 1 THROUGH 30 ARE AIRCRAFT ATTRIBUTES.  PILOTS HAVE
21          ; ATTRIBUTES 1 THROUGH 12.  WHEN PILOTS GET INTO AN AIRCRAFT THEIR
22          ; ATTRIBUTES ARE TRANSFERRED INTO AIRCRAFT ATTRIBUTES 31 THROUGH 42.
23          ; THE REMAINING AIRCRAFT ATTRIBUTES, 43 THROUGH 48, ARE MISSION
24          ; ATTRIBUTES FOR THE FLIGHT AS A WHOLE.  ONLY AIRCRAFT AND PILOTS
25          ; CARRY ATTRIBUTES.  INDIVIDUAL ATTRIBUTES ARE ENUMERATED BELOW-
26          ;
27          ; AIRCRAFT ATTRIBUTES
28          ;
29          ;  1 - SQ ID NUMBER (1-6)
30          ;  2 - TAIL NUMBER
31          ;  3 - TYPE PARKING, 0- SHELTER ON QRA, 1- SHELTER, 2- REVETMENT,
32          ;        4- WING MX REVETTEMENT, 5- NOT PARKED
33          ;  4 - # SORTIES ON DAY 1
34          ;  5 - # SORTIES ON DAY 2
35          ;  6 - # SORTIES ON DAY 3
36          ;  7 - TOTAL MINUTES ENGINE RUN TIME
37          ;  8 - TEMPORARY STORAGE FOR SERVICE DURATIONS TO INCREMENT ATRIB 7
38          ;  9 - GUN STATUS, 0- EXPENDED,UNLOADED, 1- LOADED, 2- RUNAWAY,
39          ;        3- EXPLODED WHEN FIRED
40          ; 10 - BOMBS/RX, 0- EXPENDED,UNLOADED, 1- LOADED, 2- HUNG
41          ; 11 - MISSILES(AIM OR AGM), 0- EXPENDED,UNLOADED, 1- LOADED,
42          ;        2- MALFUNCTIONED
43          ;  12 - CONFIGURATION BY AREA (1,2, OR 3)(SET IN USER1)
44          ; 13 - AIRCRAFT LOCATION ON THE AIRFIELD
45          ; 14 - AIRCRAFT GROUND TIME WITH ENGINE RUNNING
46          ; 15 - WHILE AIRCRAFT IS ON MISSION IT IS MISSION DURATION, AFTER
47          ;        LANDING IT IS SET TO FUEL REQUIRED DURING REFUELING
48          ; 16 - BATTLE DAMAGE CODE, 0- NONE, 1- LIGHT(SMALL ARMS),
49          ;        2- HEAVIER LIGHT(FRAG), 3-MODERATE(SOME STRUCTURAL DAMAGE
50          ;        BUT PROBABLY REPAIRABLE), 4- SERIOUS(DIFFICULT TO FLY AND
51          ;        A LONG REPAIR TIME OR UNABLE TO REPAIR LOCALLY), 5- SEVERE
52          ;        (CRASH LANDED, CANNOT BE REPAIRED LOCALLY IF AT ALL)
53          ; 17 - SYMPATHETIC ABORT CODE, 1- GROUND, 2- AIR, ALSO MAINTENANCE
54          ;        PROCESSING PRIORITY CODE IF AIRCRAFT DO NOT PROCEED TO
55          ;        TURNAROUND SERVICE DIRECTLY.  PRIORITY CODE FOR WING IS
56          ;        THE SUM OF THE 4 AND 5 LEVEL FAILURES. FOR MMT THE SAME.
57          ;        FOR SQUADRON MX IT IS THE SUM OF THE 2,3,4 AND 5 FAILURES.
58          ;        PROCESSING IS ACCOMPLISHED USING LOW VALUE FIRST FOR THIS
59          ;        ATTRIBUTE (FIX THE LEAST BROKE FIRST)
60          ; 18 - MAINTENANCE FAILURE CODE -A SIX DIGIT CODE WITH EACH DIGIT, IN
61          ;      ORDER, RELATED TO FAILURES IN THE SYSTEMS WHOSE NEXT TIME OF
62          ;      FAILURE IS CARRIED BY ATTRIBUTES 19 THROUGH 24. FAILURE
63          ;      OF A SYSTEM OCCURS WHEN ENGINE RUN TIME EXCEEDS THE NEXT
64          ;      TIME OF FAILURE OF A SYSTEM.  THE LEVEL OF FAILURE IS SET
65          ;      PROBABILISTICALLY.
66          ; 19 - NEXT TIME OF FAILURE (NTOF) OF SYSTEM 1 - ELECTRICAL
```

```
67          ; 20 - NTOF OF SYSTEM 2 - ENGINE/FUEL
68          ; 21 - NTOF OF SYSTEM 3 - HYDRAULICS/PNEUMATICS
69          ; 22 - NTOF OF SYSTEM 4 - AIRFRAME(INCLUDES STRUTS AND TIRES)
70          ; 23 - NTOF OF SYSTEM 5 - COMM/NAV/INSTRUMENTS/RADAR
71          ; 24 - NTOF OF SYSTEM 6 - FIRE CONTROL/WEAPONS RELEASE
72          ; 25 - CARRIES MARK TIME FOR TURNAROUND STATISTICS DAY 1
73          ; 26 - CARRIES MARK TIME FOR TURNAROUND STATISTICS DAY 2
74          ; 27 - CARRIES MARK TIME FOR TURNAROUND STATISTICS DAY 3
75          ; 28 - TOTAL MINUTES ENGINE RUN TIME DAY 1
76          ; 29 - TOTAL MINUTES ENGINE RUN TIME DAY 2
77          ; 30 - TOTAL MINUTES ENGINE RUN TIME DAY 3
78          ;
79          ; PILOT ATTTIBUTES
80          ;
81          ;  1 - SQ ID NUMBER (1-6)
82          ;  2 - PILOT ID NUMBER
83          ;  3 - PILOT STATUS, 0-PILOT, 1- FLIGHT LEAD (NON-QRA QUALIFIED),
84          ;        2- FLIGHT LEAD (QRA QUALIFIED), 3- FLIGHT LEAD ON QRA
85          ;  4 - # SORTIES ON DAY 1
86          ;  5 - # SORTIES ON DAY 2
87          ;  6 - # SORTIES ON DAY 3
88          ;  7 - MARK TIME FOR FLYING TIME DAY 1
89          ;  8 - MARK TIME FOR FLYING TIME DAY 2
90          ;  9 - MARK TIME FOR FLYING TIME DAY 3
91          ; 10 - CARRIES MARK TIME FOR PILOT GROUND TIME STATISTICS
92          ;        FOR TIME BETWEEN ENGINE SHUT DOWN AND NEXT TIME
93          ;        A PREFLIGHT IS BEGUN  DAY1
94          ; 11 - MARK TIME FOR PILOT GROUND TIME DAY 2
95          ; 12 - MARK TIME FOR PILOT GROUND TIME DAY 3
96          ;
97          ; MISSION ATTRIBUTES
98          ;
99          ; 43 - FLIGHT AIRCREW COMPOSITION, CASE I - 1 FLT LEAD, CASE II -
100         ;        2 FLT LEADS, CASE III - 3 FLT LEADS (THE SECOND FLT LEAD
101         ;        IN CASE II IS ALWAYS IN THE NUMBER 3 AIRCRAFT)
102         ; 44 - NUMBER OF AIRCRAFT IN THE FLIGHT (2 OR 3)(2 MINIMUM)
103         ; 45 - AIRCRAFT POSITION IN THE FLIGHT (1 OR LEAD, 2, OR 3, WITH A
104         ;        FLIGHT LEAD IN POSITION 1 AT ALL TIMES)
105         ; 46 - MISSION NUMBER (COUNTER GOES FROM 1 TO 46 THEN RECYCLES
106         ;        FROM 1 TO 46)
107         ; 47 - AREA TO WHICH MISSION IS SENT, 1- CLOSE, LO-LO-LO PROFILE,
108         ;        2- FURTHER, LO-LO-HIGH, 3- FARTHEST, HIGH-LO-HIGH
109         ; 48 - ALWAYS ZERO, USED TO MATCH PILOTS WITH A/C
110         ;
111         ;
112         ;    THE MAXIMUM NUMBER OF ENTRIES IN ALL FILES (MNTRY) SPECIFIED AS
113         ; 506 PROVIDES A SAFETY MARGIN FOR THE WORST CASE THE MODEL SHOULD
114         ; ENCOUNTER WITH 6 SQUADRONS OF 16 UE AIRCRAFT.
115         ;
116         GEN,TESTFIN,MANN & SHOOK 91 AFIT,1/7/82,1;
```

```
117             LIMITS,99,48,506;
118             ;
```

```
119             ; NUMBERED ACTIVITIES
120             ;    STATISTICS ARE COLLECTED ON THE FOLLOWING ACTIVITIES-
121             ;
122             ;  1 - RECONFIGURATION
123             ;  2 - REARMING
124             ;  3 - MX POST-FLIGHT
125             ;  4 - HOTPIT REFUELING
126             ;  5 - SHELTER REFUELING
127             ;  6 - TRUCK REFUELING
128             ;  7 - WGSHOP1 SERVICE
129             ;  8 - WGSHOP2 SERVICE
130             ;  9 - WGSHOP3 SERVICE
131             ; 10 - WGSHOP4 SERVICE
132             ; 11 - MMT1 SERVICE
133             ; 12 - MMT2 SERVICE
134             ; 13 - MMT3 SERVICE
135             ; 14 - MMT4 SERVICE
136             ; 15 - MMT5 SERVICE
137             ; 16 - MMT6 SERVICE
138             ; 17 - SQ11 MX SERVICE
139             ; 18 - SQ12 MX SERVICE
140             ; 19 - SQ13 MX SERVICE
141             ; 20 - SQ14 MX SERVICE
142             ; 21 - SQ21 MX SERVICE
143             ; 22 - SQ22 MX SERVICE
144             ; 23 - SQ23 MX SERVICE
145             ; 24 - SQ24 MX SERVICE
146             ; 25 - SQ31 MX SERVICE
147             ; 26 - SQ32 MX SERVICE
148             ; 27 - SQ33 MX SERVICE
149             ; 28 - SQ34 MX SERVICE
150             ; 29 - SQ41 MX SERVICE
151             ; 30 - SQ42 MX SERVICE
152             ; 31 - SQ43 MX SERVICE
153             ; 32 - SQ44 MX SERVICE
154             ; 33 - SQ51 MX SERVICE
155             ; 34 - SQ52 MX SERVICE
156             ; 35 - SQ53 MX SERVICE
157             ; 36 - SQ54 MX SERVICE
158             ; 37 - SQ61 MX SERVICE
159             ; 38 - SQ62 MX SERVICE
160             ; 39 - SQ63 MX SERVICE
161             ; 40 - SQ64 MX SERVICE
162             ; 41 - NORMAL PREFLIGHT
163             ; 42 - DELAYED PREFLIGHT
164             ; 43 - NORMAL TMA3
165             ; 44 - DELAYED TMA3
166             ; 45 - NORMAL TMA2
167             ; 46 - DELAYED TMA2
168             ; 47 - DEARM SERVICE AT DEAR
```

```
169             ; 48 - DEARM SERVICE AT DEA3
170             ; 49 - DOWNLOAD ORDNANCE AT WING MAINTENANCE.
171             ; 50 - DOWNLOAD ORDNANCE FOR MMT MAINTENANCE.
172             ;
```

```
173           ; FILE ASSIGNMENTS
174           ;
175           ;
176           ;  1 - 18  -- A/C READY POOL,PILOT READY POOL & MATCH Q'S (PER SQDN)
177           ; 19 - 20  -- QRA A/C AND PILOTS
178           ; 21       -- AWAIT MXTEAM (PREFLIGHT)
179           ; 22 - 44  -- MATCH Q'S FOR LAUNCH PROCESS
180           ; 45 - 46  -- AWAIT RUNWAY (TAKEOFF)
181           ; 47       -- AWAIT RUNWAY (LANDING)
182           ; 48 - 49  -- AWAIT DEARMING
183           ; 50       -- AWAIT HOTPIT REFUELING
184           ; 51       -- AWAIT MXTEAM (TURNAROUND SERVICE)
185           ; 52       -- AWAIT REARMING
186           ; 53       -- AWAIT FUEL TRUCK
187           ; 54       -- GATE TO CLOSE IF OUT OF FUEL
188           ; 55 - 57  -- MATCH AFTER TURNAROUND SERVICING
189           ; 58 - 61  -- WING MX SHOPS
190           ; 62       -- WING MX QUEUE
191           ; 63 - 68  -- MMT MX UNIT
192           ; 69       -- MMT MX QUEUE
193           ; 70 - 73  -- SQDN 1 MX SHOPS
194           ; 74 - 77  -- SQDN 2 MX SHOPS
195           ; 78 - 81  -- SQDN 3 MX SHOPS
196           ; 82 - 85  -- SQDN 4 MX SHOPS
197           ; 86 - 89  -- SQDN 5 MX SHOPS
198           ; 90 - 93  -- SQDN 6 MX SHOPS
199           ; 94 - 97  -- MATCH AFTER SQDN MX SERVICE
200           ; 98       -- AWAIT MXTEAM (MX CONTROL)
201           ; 99       -- JUNK FILE (A/C ATTRITED, CRASHED, OR SCRAPPED)
202           ;
```

```
203             ; GLOBAL VARIABLE ASSIGNMENTS
204             ;
205             ;
206             ; 1 - MISSION VARIABLES
207             ;
208             ;     XX(1)... XX(46) - MISSION STATUS (USED TO DETERMINE CURRENT MAIN-
209             ;               TENANCE STATUS OF A/C IN A PARTICULAR FLIGHT DURING START,
210             ;               TMA, TAKEOFF, AND REJOIN.  USED TO ASSIGN MISSION DURA-
211             ;               TION TO EACH A/C IN A PARTICULAR FLIGHT (WHILE AIRBORNE).
212             ;     XX(47) - MISSION NUMBER (SET BY SCHEDULER).
213             ;     XX(48) - REQUIRED EXTERNAL TANK CONFIGURATION (SET BY SCHEDULER).
214             ;     XX(49) - MISSION CREW COMPOSITION (SET BY ORGANPT).
215             ;     XX(97) - AREA MISSION IS GOING TO (SET BY SCHEDULER).
216             ;
217             ; 2 - CREATION VARIABLES
218             ;
219             ;     XX(56) - A/C CREATION COUNTER
220             ;     XX(57) - NUMBER OF A/C TO BE CREATED PER SQUADRON.
221             ;     XX(58) - NUMBER OF PILOTS TO BE CREATED PER SQUADRON.
222             ;     XX(59) - INITIAL CONFIGURATION
223             ;     XX(61) - NUMBER OF PILOT/AIRCRAFT ON QRA
224             ;     XX(62) - NUMBER OF QRA QUALIFIED PILOTS/SQUADRON
225             ;     XX(63) - NUMBER OF FLIGHT LEAD QUALIFIED PILOTS/SQUADRON
226             ;     XX(64) - PERCENT OF A/C INITIALLY OPERATIONAL
227             ;     XX(70) - PILOT ID NUMBER COUNTER
228             ;     XX(71) - PILOT STATUS COUNTER
229             ;     XX(72) - A/C TAIL NUMBER COUNTER
230             ;     XX(73) - A/C FLIGHT POSITION COUNTER (RESET TO ZERO BY SCHEDULER).
231             ;     XX(74)... XX(79) - A/C QRA STATUS COUNTERS, SQ1 THRU SQ6.
232             ;
233             ; 3 - OPERATIONAL VARIABLES
234             ;
235             ;     XX(65) - PROBABILITY OF A/C DELAY AT PILOT PREFLIGHT
236             ;     XX(66) - PROBABILITY OF A/C DELAY AT START
237             ;     XX(67) - PROBABILITY OF A/C DELAY AT TMA
238             ;     XX(68) - PROBABILITY OF FLT DELAY AT TAKEOFF
239             ;     XX(69) - PROBABILITY OF A/C DELAY AT REJOIN
240             ;
241             ; 4 - NETWORK VARIABLES
242             ;
243             ;     XX(50)... XX(55) - RESUPPLY REQUEST, SQ1 THRU SQ6
244             ;     XX(80)... XX(91) - PILOT/AIRCRAFT FORMATION
245             ;     XX(92) - MISSION FLIGHT TIME
246             ;     XX(93) - MMT POOL SWITCH
247             ;     XX(95) - DUMMY VARIABLE FOR FUNCTION CALLS
248             ;     XX(96) - EXECUTIVE NETWORK SWITCH
249             ;     XX(98) - MASTER CLOCK DELAY TIME
250             ;
251             ; 5 - RESPONSE VARIABLE
252             ;
```

```
253             ;       XX(94) - TOTAL EFFECTIVE SORTIES FLOWN
254             ;
255             ; 6 - OTHER VARIABLES
256             ;
257             ;       XX(60) - INITIAL POL SUPPLY
258             ;       XX(99) - UNUSED
259             ;       XX(100)- COUNTER FOR NUMBER OF A/C EXPERIENCING A MX FAILURE.
260             ;
```

```
261          ; MAINTENANCE PRIORITY PROCESSING
262          ;    THE PROCESSING OF A/C IN MAINTENANCE IS BASED ON LOW
263          ; VALUE FIRST OF ATTRIBUTE 17.  THE WAY THE VALUE IS COMPUTED
264          ; IS EXPLAINED IN THE MAINTENANCE SECTIONS.
265          ;
266          PRIORITY/62,LVF(17);
267          PRIORITY/69,LVF(17);
268          PRIORITY/70,LVF(17);
269          PRIORITY/71,LVF(17);
270          PRIORITY/72,LVF(17);
271          PRIORITY/73,LVF(17);
272          PRIORITY/74,LVF(17);
273          PRIORITY/75,LVF(17);
274          PRIORITY/76,LVF(17);
275          PRIORITY/77,LVF(17);
276          PRIORITY/78,LVF(17);
277          PRIORITY/79,LVF(17);
278          PRIORITY/80,LVF(17);
279          PRIORITY/81,LVF(17);
280          PRIORITY/82,LVF(17);
281          PRIORITY/83,LVF(17);
282          PRIORITY/84,LVF(17);
283          PRIORITY/85,LVF(17);
284          PRIORITY/86,LVF(17);
285          PRIORITY/87,LVF(17);
286          PRIORITY/88,LVF(17);
287          PRIORITY/89,LVF(17);
288          PRIORITY/90,LVF(17);
289          PRIORITY/91,LVF(17);
290          PRIORITY/92,LVF(17);
291          PRIORITY/93,LVF(17);
292          ;
```

```
 293           ; RESOURCE INITIALIZATION
294           ;    EACH RESOURCE IS SET TO ITS INITIAL CAPACITY AND THE PRIORITY
295           ; FOR USERS IS SPECIFIED BY THE FILE NUMBER ORDER.  FOR EXAMPLE,
296           ; MXTEAM(96),21,51,98 SPECIFIES 9o CREW CHIEFS AVAILABLE WITH PRI-
297           ; ORITY TO PREFLIGHT, THEN ENGINE SHUT DOWN AND FINALLY AN ALLOCA-
298           ; TION AT MAINTENANCE CONTROL.
299           ;
300           NETWORK;
301                RESOURCE/WCSHOP1(0),58;
302                RESOURCE/WCSHOP2(0),59;
303                RESOURCE/WCSHOP3(0),60;
304                RESOURCE/WCSHOP4(0),61;
305                RESOURCE/MMT1(0),63;
306                RESOURCE/MMT2(0),64;
307                RESOURCE/MMT3(0),65;
308                RESOURCE/MMT4(0),66;
309                RESOURCE/MMT5(0),67;
310                RESOURCE/MMT6(0),68;
311                RESOURCE/MXTEAM(0),21,51,98;
312                RESOURCE/SQ1MX1(0),70;
313                RESOURCE/SQ1MX2(0),71;
314                RESOURCE/SQ1MX3(0),72;
315                RESOURCE/SQ1MX4(0),73;
316                RESOURCE/SQ2MX1(0),74;
317                RESOURCE/SQ2MX2(0),75;
318                RESOURCE/SQ2MX3(0),76;
319                RESOURCE/SQ2MX4(0),77;
320                RESOURCE/SQ3MX1(0),78;
321                RESOURCE/SQ3MX2(0),79;
322                RESOURCE/SQ3MX3(0),80;
323                RESOURCE/SQ3MX4(0),81;
324                RESOURCE/SQ4MX1(0),82;
325                RESOURCE/SQ4MX2(0),83;
326                RESOURCE/SQ4MX3(0),84;
327                RESOURCE/SQ4MX4(0),85;
328                RESOURCE/SQ5MX1(0),86;
329                RESOURCE/SQ5MX2(0),87;
330                RESOURCE/SQ5MX3(0),88;
331                RESOURCE/SQ5MX4(0),89;
332                RESOURCE/SQ6MX1(0),90;
333                RESOURCE/SQ6MX2(0),91;
334                RESOURCE/SQ6MX3(0),92;
335                RESOURCE/SQ6MX4(0),93;
336                RESOURCE/REARM(0),52;
337                RESOURCE/REFUEL(0),53;
338                RESOURCE/DEARM(0),48,49;
339                RESOURCE/RUNWAY(0),47,45,46;
340                RESOURCE/HOTPIT(0),50;
341           ;
```

```
342          ; GATE INITIALIZATION
343          ;    EACH GATE HAS AN INITIAL POSITION, OPEN OR CLOSED.  GATES ARE USED
344          ; TO KEEP AIRCRAFT IN AN AWAIT NODE UNTIL THE OCCURENCE OF SOME EVENT
345          ; WHICH MEANS THEY COULD POSIBLY ACQUIRE A RESOURCE THEY NEED -- AT
346          ; THAT POINT AN OPEN GATE NODE IS USED TO ALLOW THE AIRCRAFT TO FLOW
347          ; ON.  IN THE CASE OF FUEL THE GATE IS INITIALLY OPEN AND IT IS CLOSED
348          ; WHEN THE FUEL SUPPLY IS EXHAUSTED.
349          ;
350               GATE/JUNK,CLOSE,99;
351               GATE/FUELAVAL,OPEN,54;
352               GATE/MMTPOOL,OPEN,69;
353               GATE/WCPOOL,OPEN,62;
354               GATE/RDYPOOL1,CLOSE,1;
355               GATE/RDYPOOL2,CLOSE,4;
356               GATE/RDYPOOL3,CLOSE,7;
357               GATE/RDYPOOL4,CLOSE,10;
358               GATE/RDYPOOL5,CLOSE,13;
359               GATE/RDYPOOL6,CLOSE,16;
360               GATE/QRAPOOL,CLOSE,19;
361               GATE/PILOTQRA,CLOSE,20;
362          ;
```

```
; AIRCRAFT AND PILOT CREATION
;    AIRCRAFT ARE CREATED AT TIME 0.  IF RESUPPLY IS REQUIRED (REPLACE-
; MENT SQUADRONS), ADDITIONAL AIRCRAFT ARE CREATED ON DAY 2 AND DAY
; 3.  THE USER CAN SPECIFY THE NUMBER OF AIRCRAFT TO BE USED FOR
; EACH SQUADRON (UE) IN THE FORTRAN CODE (MAX OF 50).  THIS NUMBER
; APPLIES TO ALL SQUADRONS AND TO ALL REPLACEMENT SQUADRONS SCHEDULED BY
; SUBROUTINE RESUPLY.  THE DECISION TO RESUPPLY IS MADE BY THE
; PROGRAM BASED ON THE NUMBER OF AIRCRAFT THE USER SPECIFIES
; FOR LIMITAC.  THIS IS THE MINIMUM NUMBER OF OPERATIONAL A/C
; AT WHICH THE USER FEELS THE SQUADRON CAN FUNCTION EFFECTIVELY.
; THE TIME OF RESUPPLY CAN BE ALTERED. CURRENTLY, REPLACEMENT
; SQUADRONS ARE SCHEDULED TO ARRIVE AT 1800.0 MINUTES AND 3240.0
; MINUTES INTO THE RUN (MID-DAY THE FOLLOWING DAY), IF REQUIRED.
;    THE INITIAL AIRCRAFT ARE PROCESSED DIRECTLY THROUGH THE NODES
; IN THIS SECTION.  THEY GET THEIR INITIAL VALUES SET IN THIS
; AREA.  FOR EXAMPLE, NTOF FOR EACH OF THE SIX SYSTEMS BASED ON
; A PROBABILISTIC DRAW FROM A DISTRIBUTION BASED ON THE MEAN TIME
; BETWEEN FAILURE (MTBF) FOR THAT SYSTEM.  A UNIQUE TAIL NUMBER IS
; ASSIGNED AS WELL AS A SQUADRON NUMBER AND A PARKING SPACE.  AIR-
; CRAFT WHICH DON'T REQUIRE MAINTENANCE INITIALLY GO ON QUICK
; REACTION ALERT (QRA) OR TO THE READY POOL. THE NUMBER WHICH GO
; ON QRA IS USER SELECTABLE IN THE FORTRAN.  THE PERCENTAGE OF A/C
; INITIALLY OPERATIONALLY READY (OR) IS USER SELECTABLE.  THOSE A/C
; NOT INITIALLY OR GO TO MX.
; REPLACEMENT SQUADRON AIRCRAFT GO THROUGH THE SAME INITIAL
; ASSIGNMENT ROUTINE BUT ARE THEN BRANCHED THROUGH RESC TO
; PICK UP A PILOT BEFORE GOING TO APPROACH FOR LANDING.
;    THE INITIAL NUMBER OF PILOTS AND THEIR QUALIFICATIONS (FLIGHT
; LEAD/QRA QUALIFIED) IS USER SPECIFIED FOR UP TO 75 PILOTS PER
; SQUADRON.
;    WHEN A MISSION IS SCHEDULED, SCHEDULER WILL OPEN THE READY POOL
; GATE IF ENOUGH AIRCRAFT ARE IN THE READY POOL TO FORM A FLIGHT.
; THREE AIRCRAFT OF THE CORRECT CONFIGURATION ARE ASSIGNED TO A
; PARTICULAR MISSION NUMBER AND A PILOT OF PROPER QUALIFICATION
; IS SCHEDULED AGAINST THE AIRCRAFT.  MISSION ATTRIBUTES ARE SET.
; THE AIRCRAFT AND PILOT THEN PROCEED TO PILOT PREFLIGHT.
;
;                                        SQ1 A/C GENERATION ROUTINE
;
;
     CREATE,0,0,,50,1;                  INITIAL A/C GENERATION FOR SQ1
       ACT,,XX(50).EQ.1,NAL1;
       ACT,,XX(50).EQ.0,TERM;
     CREATE,0,1800,,50,1;               IF DAY02 REPLACEMENT SQ1 REQUIRED
       ACT,,XX(50).EQ.1,NAL1;
       ACT,,XX(50).EQ.0,TERM;
     CREATE,0,3240,,50,1;               IF DAY03 REPLACEMENT SQ1 REQUIRED
       ACT,,XX(50).EQ.1,NAL1;
       ACT,,XX(50).EQ.0,TERM;
;
```

```
413             NAL1 ASSIGN,XX(56)=XX(56)+1,1;
414                     ACT,,XX(56).GT.XX(57).AND.
415                          TNOW.LT.0.1,TERM;
416                     ACT,,XX(56).GT.XX(57),RESC;
417                     ACT,,,ASQ1;
418                ;                                   A/C IDENTIFICATION (SQ,TAIL #)
419                ;                                   AND STATUS CODES.
420             ASQ1 ASSIGN,XX(72)=XX(72)+1,
421                          ATRIB(26)=0,ATRIB(27)=0,
422                          ATRIB(1)=1, ATRIB(2)=USERF(18),
423                          ATRIB(7)=USERF(137),
424                          ATRIB(12)=XX(59),ATRIB(14)=0,ATRIB(15)=0,
425                          ATRIB(16)=0,ATRIB(17)=0,ATRIB(18)=0;
426                ;
427                ;                                   A/C STATISTICS AND
428                ;                                   NEXT TIME OF FAILURE, BY SYSTEM.
429                     ASSIGN,
430                          ATRIB(4)=0, ATRIB(5)=0, ATRIB(6)=0,
431                          ATRIB(28)=0,ATRIB(29)=0,ATRIB(30)=0,
432                          ATRIB(19)=USERF(131),ATRIB(20)=USERF(132),
433                          ATRIB(21)=USERF(133),ATRIB(22)=USERF(134),
434                          ATRIB(23)=USERF(135),ATRIB(24)=USERF(136),1;
435                     ACT,,XX(50).EQ.1.AND.
436                          TNOW.GT.0.1,RESC;          REPLACEMENT A/C
437                     ACT,,DRAND.GT.XX(64),MXIS;      A/C IN MAINT INITIALLY
438                     ACT,,,ASC1;                     GENERATED OPERATIONAL A/C
439                ;
440                ;                                   INITIALIZE CONFIGURATION
441             ASC1 ASSIGN,ATRIB(3)=USERF(21),
442                          ATRIB(9)=1, ATRIB(10)=1,ATRIB(11)=0,
443                          XX(74)=XX(74)+1,1;
444                     ACT,,XX(74).LE.XX(61),QRA;      SELECT QRA A/C (FIRST TIME ONLY)
445                     ACT,,XX(74).GT.XX(61);          OPERATIONAL AIRCRAFT
446                ;
447                ;
448                ;
449                ;
450                ;                                   FORM A FLIGHT OF THREE
451                ;
452                ;                                   GATE OPENED BY SCHEDULER
453                ;
454             ARP1 AWAIT(1),RDYPOOL1,1;              READY POOL SQ1
455                     ACT,,ATRIB(12).NE.XX(48),CA11; IF WRONG CONFIG, RETURN TO RDYPOOL
456                     ACT,,ATRIB(12).EQ.XX(48),ACC1;
457             CA11 CLOSE,RDYPOOL1,1;
458                     ACT,,,ARP1;
459                ;
460             ACC1 ASSIGN,XX(73)=XX(73)+1,
461                      ATRIB(43)=XX(49),ATRIB(44)=3,
462                      ATRIB(45)=XX(73),ATRIB(46)=XX(47),
```

```
                ATRIB(47)=XX(97),1;          ASSIGN MISSION INFO
             ACT,,XX(73).GT.3,ARP1;          RETURN REST TO READY POOL
             ACT,,XX(73).LE.3,GAT1;
;
 GAT1 CLOSE,RDYPOOL1,1;
             ACT,,,ACS1;                     TO MATCH WITH PILOT
;
;
;
 ACS1 QUEUE(3),,,,APM1;                      TO MATCH OF A/C & PILOTS
;
;
;
;
          CREATE,0,0,,75,1;                  INITIAL PILOT GENERATION FOR SQ1
             ACT,,XX(50).EQ.1,ASP1;
             ACT,,XX(50).EQ.0,TERM;
;
;                                           PILOT ID (SQ,ID#) AND
;                                           PILOT ACTIVITY STATISTICS.
 ASP1 ASSIGN,XX(70)=XX(70)+1,XX(71)=XX(71)+1,
                 ATRIB(1)=1,ATRIB(2)=XX(70),
                 ATRIB(3)=USERF(11),
                 ATRIB(4)=0,ATRIB(5)=0,
                 ATRIB(6)=0,ATRIB(11)=0,
                 ATRIB(12)=0,
                 ATRIB(8)=0,ATRIB(9)=0,1;
             ACT,,XX(71).EQ.75,QA1;
             ACT,,XX(71).LE.XX(61),PQRP;
             ACT,,XX(71).LE.XX(58),PL1;
             ACT,,,TERM;
;
 QA1  ASSIGN,XX(71)=0,XX(56)=0,1;                AUTO RESET OF COUNTER
             ACT,,XX(58).EQ.75,PL1;
             ACT,,,TERM;
;
;
 PL1  QUEUE(2),,,,APM1;                      TO A/C/PILOT MATCH
;
 APM1 MATCH,48,PL1/APLT,ACS1/ASAC;           ASSIGN A/C TO PILOTS
;
;
;
;
;
;                                           SQ2 A/C GENERATION ROUTINE
;
;
          CREATE,0,.0001,,50,1;             INITIAL A/C GENERATION FOR SQ2
             ACT,,XX(51).EQ.1,NAL2;
```

```
513                 ACT,,XX(51).EQ.Ø,TERM;
514               CREATE,Ø,1800.0001,,50,1;        IF DAYØ2 REPLACEMENT SQ2 REQUIRE
515                 ACT,,XX(51).EQ.1,NAL2;
516                 ACT,,XX(51).EQ.Ø,TERM;
517               CREATE,Ø,3240.0001,,50,1;        IF DAYØ3 REPLACEMENT SQ2 REQUIRE
518                 ACT,,XX(51).EQ.1,NAL2;
519                 ACT,,XX(51).EQ.Ø,TERM;
520           ;
521            NAL2 ASSIGN,XX(56)=XX(56)+1,1;
522                 ACT,,XX(56).GT.XX(57).AND.
523                      TNOW.LT.Ø.1,TERM;
524                 ACT,,XX(56).GT.XX(57),RESC;
525                 ACT,,,ASQ2;
526           ;                                    A/C IDENTIFICATION (SQ,TAIL #)
527           ;                                    AND STATUS CODES.
528            ASQ2 ASSIGN,XX(72)=XX(72)+1,
529                      ATRIB(26)=Ø,ATRIB(27)=Ø,
530                      ATRIB(1)=2, ATRIB(2)=USERF(18),
531                      ATRIB(7)=USERF(137),
532                      ATRIB(12)=XX(59),ATRIB(14)=Ø,ATRIB(15)=Ø,
533                      ATRIB(16)=Ø,ATRIB(17)=Ø,ATRIB(18)=Ø;
534           ;
535           ;                                    A/C STATISTICS AND
536           ;                                    NEXT TIME OF FAILURE, BY SYSTEM.
537                ASSIGN,
538                      ATRIB(4)=Ø, ATRIB(5)=Ø, ATRIB(6)=Ø,
539                      ATRIB(28)=Ø,ATRIB(29)=Ø,ATRIB(30)=Ø,
540                      ATRIB(19)=USERF(131),ATRIB(20)=USERF(132),
541                      ATRIB(21)=USERF(133),ATRIB(22)=USERF(134),
542                      ATRIB(23)=USERF(135),ATRIB(24)=USERF(136),1;
543                   ACT,,XX(51).EQ.1.AND.
544                        TNOW.GT.Ø.1,RESC;       REPLACEMENT A/C
545                   ACT,,DRAND.GT.XX(64),MXIS;    A/C IN MAINT INITIALLY
546                   ACT,,,ASC2;                   GENERATED OPERATIONAL A/C
547           ;
548           ;                                    INITIALIZE CONFIGURATION
549            ASC2 ASSIGN,ATRIB(3)=USERF(21),
550                      ATRIB(9)=1, ATRIB(10)=1,ATRIB(11)=Ø,
551                      XX(75)=XX(75)+1,1;
552                   ACT,,XX(75).LE.XX(61),QRA;   SELECT QRA A/C (FIRST TIME ONLY)
553                   ACT,,XX(75).GT.XX(61);       OPERATIONAL AIRCRAFT
554           ;
555           ;
556           ;
557           ;
558           ;                                    FORM A FLIGHT OF THREE
559           ;
560           ;                                    GATE OPENED BY SCHEDULER
561           ;
562            ARP2 AWAIT(4),RDYPOOL2,1;           READY POOL SQ2
```

```
563                 ACT,,ATRIB(12).NE.XX(48),CA12; IF WRONG CONFIG, RETURN TO RDYPOOL
564                 ACT,,ATRIB(12).EQ.XX(48),ACC2;
565               CA12 CLOSE,RDYPOOL2,1;
566                 ACT,,,ARP2;
567               ;
568               ACC2 ASSIGN,XX(73)=XX(73)+1,
569                    ATRIB(43)=XX(49),ATRIB(44)=3,
570                    ATRIB(45)=XX(73),ATRIB(46)=XX(47),
571                    ATRIB(47)=XX(97),1;          ASSIGN MISSION INFO
572                 ACT,,XX(73).GT.3,ARP2;          RETURN REST TO READY POOL
573                 ACT,,XX(73).LE.3,CAT2;
574               ;
575               CAT2 CLOSE,RDYPOOL2,1;
576                 ACT,,,ACS2;                     TO MATCH WITH PILOT
577               ;
578               ;
579               ;
580               ACS2 QUEUE(6),,,,APM2;            TO MATCH OF A/C & PILOTS
581               ;
582               ;
583               ;
584               ;
585                    CREATE,0,.0001,,75,1;        INITIAL PILOT GENERATION FOR SQ2
586                   ACT,,XX(51).EQ.1,ASP2;
587                   ACT,,XX(51).EQ.0,TERM;
588               ;
589               ;                                 PILOT ID (SQ,ID#) AND
590               ;                                 PILOT ACTIVITY STATISTICS.
591               ASP2 ASSIGN,XX(70)=XX(70)+1,XX(71)=XX(71)+1,
592                        ATRIB(1)=2,ATRIB(2)=XX(70),
593                        ATRIB(3)=USERF(11),
594                        ATRIB(4)=0,ATRIB(5)=0,
595                        ATRIB(6)=0,ATRIB(11)=0,
596                        ATRIB(12)=0,
597                        ATRIB(8)=0,ATRIB(9)=0,1;
598                 ACT,,XX(71).EQ.75,QA2;
599                 ACT,,XX(71).LE.XX(61),PQRP;
600                 ACT,,XX(71).LE.XX(58),PL2;
601                 ACT,,,TERM;
602               ;
603               QA2  ASSIGN,XX(71)=0,XX(56)=0,1;          AUTO RESET OF COUNTER.
604                 ACT,,XX(58).EQ.75,PL2;
605                 ACT,,,TERM;
606               ;
607               ;
608               PL2  QUEUE(5),,,,APM2;            TO A/C/PILOT MATCH
609               ;
610               APM2 MATCH,48,PL2/APLT,ACS2/ASAC;     ASSIGN A/C TO PILOTS
611               ;
612               ;
```

```
;
;
;                                       SQ3 A/C GENERATION ROUTINE
;
;
     CREATE,0,.0002,,50,1;              INITIAL A/C GENERATION FOR SQ3
       ACT,,XX(52).EQ.1,NAL3;
       ACT,,XX(52).EQ.0,TERM;
     CREATE,0,1800.0002,,50,1;          IF DAY02 REPLACEMENT SQ3 REQUIRE
       ACT,,XX(52).EQ.1,NAL3;
       ACT,,XX(52).EQ.0,TERM;
     CREATE,0,3240.0002,,50,1;          IF DAY03 REPLACEMENT SQ3 REQUIRE
       ACT,,XX(52).EQ.1,NAL3;
       ACT,,XX(52).EQ.0,TERM;
;
NAL3 ASSIGN,XX(56)=XX(56)+1,1;
       ACT,,XX(56).GT.XX(57).AND.
            TNOW.LT.0.1,TERM;
       ACT,,XX(56).GT.XX(57),RESC;
       ACT,,,ASQ3;
;                                       A/C IDENTIFICATION (SQ,TAIL #)
;                                       AND STATUS CODES.
ASQ3 ASSIGN,XX(72)=XX(72)+1,
            ATRIB(26)=0,ATRIB(27)=0,
            ATRIB(1)=3, ATRIB(2)=USERF(18),
            ATRIB(7)=USERF(137),
            ATRIB(12)=XX(59),ATRIB(14)=0,ATRIB(15)=0,
            ATRIB(16)=0,ATRIB(17)=0,ATRIB(18)=0;
;
;                                       A/C STATISTICS AND
;                                       NEXT TIME OF FAILURE, BY SYSTEM.
     ASSIGN,
            ATRIB(4)=0, ATRIB(5)=0, ATRIB(6)=0,
            ATRIB(28)=0,ATRIB(29)=0,ATRIB(30)=0,
            ATRIB(19)=USERF(131),ATRIB(20)=USERF(132),
            ATRIB(21)=USERF(133),ATRIB(22)=USERF(134),
            ATRIB(23)=USERF(135),ATRIB(24)=USERF(136),1;
       ACT,,XX(52).EQ.1.AND.
            TNOW.GT.0.1,RESC;           REPLACEMENT A/C
       ACT,,DRAND.GT.XX(54),MXIS;        A/C IN MAINT INITIALLY
       ACT,,,ASC3;                      GENERATED OPERATIONAL A/C
;
;                                       INITIALIZE CONFIGURATION
ASC3 ASSIGN,ATRIB(3)=USERF(21),
            ATRIB(9)=1, ATRIB(10)=1,ATRIB(11)=0,
            XX(76)=XX(76)+1,1;
       ACT,,XX(76).LE.XX(61),QRA;       SELECT QRA A/C (FIRST TIME ONLY)
       ACT,,XX(76).GT.XX(61);           OPERATIONAL AIRCRAFT
;
```

```
 663            ;
 664            ;
665             ;
666             ;                                   FORM A FLIGHT OF THREE
667             ;
668             ;                                   GATE OPENED BY SCHEDULER
669             ;
670             ARP3 AWAIT(7),RDYPOOL3,1;           READY POOL SQ3
671                  ACT,,ATRIB(12).NE.XX(48),GA13; IF WRONG CONFIG, RETURN TO RDYPOOL
672                  ACT,,ATRIB(12).EQ.XX(48),ACC3;
673             GA13 CLOSE,RDYPOOL3,1;
674                  ACT,,,ARP3;
675             ;
676             ACC3 ASSIGN,XX(73)=XX(73)+1,
677                   ATRIB(43)=XX(49),ATRIB(44)=3,
678                   ATRIB(45)=XX(73),ATRIB(46)=XX(47),
679                   ATRIB(47)=XX(97),1;           ASSIGN MISSION INFO
680                  ACT,,XX(73).GT.3,ARP3;         RETURN REST TO READY POOL
681                  ACT,,XX(73).LE.3,GAT3;
682             ;
683             GAT3 CLOSE,RDYPOOL3,1;
684                  ACT,,,ACS3;                    TO MATCH WITH PILOT
685             ;
686             ;
687             ;
688             ACS3 QUEUE(9),,,,APM3;              TO MATCH OF A/C & PILOTS
689             ;
690             ;
691             ;
692             ;
693                  CREATE,0,.0002,,75,1;          INITIAL PILOT GENERATION FOR SQ3
694                  ACT,,XX(52).EQ.1,ASP3;
695                  ACT,,XX(52).EQ.0,TERM;
696             ;
697             ;                                   PILOT ID (SQ,ID#) AND
698             ;                                   PILOT ACTIVITY STATISTICS.
699             ASP3 ASSIGN,XX(70)=XX(70)+1,XX(71)=XX(71)+1,
700                       ATRIB(1)=3,ATRIB(2)=XX(70),
701                       ATRIB(3)=USERF(11),
702                       ATRIB(4)=0,ATRIB(5)=0,
703                       ATRIB(6)=0,ATRIB(11)=0,
704                       ATRIB(12)=0,
705                       ATRIB(8)=0,ATRIB(9)=0,1;
706                  ACT,,XX(71).EQ.75,QA3;
707                  ACT,,XX(71).LE.XX(61),PQRP;
708                  ACT,,XX(71).LE.XX(58),PL3;
709                  ACT,,,TERM;
710             ;
711             QA3  ASSIGN,XX(71)=0,XX(56)=0,1;          AUTO RESET OF COUNTER
712                  ACT,,XX(58).EQ.75,PL3;
```

```
 713                 ACT,,,TERM;
 714           ;
 715           ;
 716           PL3  QUEUE(8),,,,APM3;                TO A/C/PILOT MATCH
 717           ;
 718           APM3 MATCH,48,PL3/APLT,ACS3/ASAC;      ASSIGN A/C TO PILOTS
 719           ;
 720           ;
 721           ;
 722           ;
 723           ;
 724           ;                                      SQ4 A/C GENERATION ROUTINE
 725           ;
 726           ;
 727                CREATE,0,.0003,,50,1;             INITIAL A/C GENERATION FOR SQ4
 728                   ACT,,XX(53).EQ.1,NAL4;
 729                   ACT,,XX(53).EQ.0,TERM;
 730                CREATE,0,1800.0003,,50,1;         IF DAY02 REPLACEMENT SQ4 REQUIRE
 731                   ACT,,XX(53).EQ.1,NAL4;
 732                   ACT,,XX(53).EQ.0,TERM;
 733                CREATE,0,3240.0003,,50,1;         IF DAY03 REPLACEMENT SQ4 REQUIRE
 734                   ACT,,XX(53).EQ.1,NAL4;
 735                   ACT,,XX(53).EQ.0,TERM;
 736           ;
 737           NAL4 ASSIGN,XX(56)=XX(56)+1,1;
 738                   ACT,,XX(56).GT.XX(57).AND.
 739                        TNOW.LT.0.1,TERM;
 740                   ACT,,XX(56).GT.XX(57),RESC;
 741                   ACT,,,ASQ4;
 742           ;                                      A/C IDENTIFICATION (SQ,TAIL #)
 743           ;                                      AND STATUS CODES.
 744           ASQ4 ASSIGN,XX(72)=XX(72)+1,
 745                       ATRIB(26)=0,ATRIB(27)=0,
 746                       ATRIB(1)=4, ATRIB(2)=USERF(18),
 747                       ATRIB(7)=USERF(137),
 748                       ATRIB(12)=XX(59),ATRIB(14)=0,ATRIB(15)=0,
 749                       ATRIB(16)=0,ATRIB(17)=0,ATRIB(18)=0;
 750           ;
 751           ;                                      A/C STATISTICS AND
 752           ;                                      NEXT TIME OF FAILURE, BY SYSTEM.
 753                ASSIGN,
 754                       ATRIB(4)=0, ATRIB(5)=0, ATRIB(6)=0,
 755                       ATRIB(28)=0,ATRIB(29)=0,ATRIB(30)=0,
 756                       ATRIB(19)=USERF(131),ATRIB(20)=USERF(132),
 757                       ATRIB(21)=USERF(133),ATRIB(22)=USERF(134),
 758                       ATRIB(23)=USERF(135),ATRIB(24)=USERF(136),1;
 759                   ACT,,XX(53).EQ.1.AND.
 760                        TNOW.GT.0.1,RESC;         REPLACEMENT A/C
 761                   ACT,,DRAND.GT.XX(64),MXIS;     A/C IN MAINT INITIALLY
 762                   ACT,,,ASC4;                    GENERATED OPERATIONAL A/C
```

```
763             ;
764             ;                                       INITIALIZE CONFIGURATION
765             ASC4 ASSIGN,ATRIB(3)=USERF(21),
766                          ATRIB(9)=1, ATRIB(10)=1,ATRIB(11)=0,
767                          XX(77)=XX(77)+1,1;
768                       ACT,,XX(77).LE.XX(61),QRA;    SELECT QRA A/C (FIRST TIME ONLY)
769                       ACT,,XX(77).GT.XX(61);        OPERATIONAL AIRCRAFT
770             ;
771             ;
772             ;
773             ;
774             ;                                       FORM A FLIGHT OF THREE
775             ;
776             ;                                       GATE OPENED BY SCHEDULER
777             ;
778             ARP4 AWAIT(10),RDYPOOL4,1;            READY POOL SQ4
779                    ACT,,ATRIB(12).NE.XX(48),CA14; IF WRONG CONFIG, RETURN TO RDYPOOL
780                    ACT,,ATRIB(12).EQ.XX(48),ACC4;
781             CA14 CLOSE,RDYPOOL4,1;
782                    ACT,,,ARP4;
783             ;
784             ACC4 ASSIGN,XX(73)=XX(73) + 1,
785                     ATRIB(43)=XX(49),ATRIB(44)=3,
786                     ATRIB(45)=XX(73),ATRIB(46)=XX(47),
787                     ATRIB(47)=XX(97),1;          ASSIGN MISSION INFO
788                    ACT,,XX(73).GT.3,ARP4;         RETURN REST TO READY POOL
789                    ACT,,XX(73).LE.3,CAT4;
790             ;
791             CAT4 CLOSE,RDYPOOL4,1;
792                    ACT,,,ACS4;                    TO MATCH WITH PILOT
793             ;
794             ;
795             ;
796             ACS4 QUEUE(12),,,,APM4;               TO MATCH OF A/C & PILOTS
797             ;
798             ;
799             ;
800             ;
801                  CREATE,0,.0003,,75,1;            INITIAL PILOT GENERATION FOR SQ4
802                    ACT,,XX(53).EQ.1,ASP4;
803                    ACT,,XX(53).EQ.0,TERM;
804             ;
805             ;                                       PILOT ID (SQ,ID#) AND
806             ;                                       PILOT ACTIVITY STATISTICS.
807             ASP4 ASSIGN,XX(70)=XX(70)+1,XX(71)=XX(71)+1,
808                          ATRIB(1)=4,ATRIB(2)=XX(70),
809                          ATRIB(3)=USERF(11),
810                          ATRIB(4)=0,ATRIB(5)=0,
811                          ATRIB(6)=0,ATRIB(11)=0,
812                          ATRIB(12)=0,
```

```
                    ATRIB(8)=0,ATRIB(9)=0,1;
               ACT,,XX(71).EQ.75,QA4;
               ACT,,XX(71).LE.XX(61),PQRP;
               ACT,,XX(71).LE.XX(58),PL4;
               ACT,,,TERM;
          ;
          QA4  ASSIGN,XX(71)=0,XX(56)=0,1;              AUTO RESET OF COUNTER
               ACT,,XX(58).EQ.75,PL4;
               ACT,,,TERM;
          ;
          ;
          PL4  QUEUE(11),,,,APM4;                       TO A/C/PILOT MATCH
          ;
          APM4 MATCH,48,PL4/APLT,ACS4/ASAC;             ASSIGN A/C TO PILOTS
          ;
          ;
          ;
          ;
          ;
          ;                                             SQ5 A/C GENERATION ROUTINE
          ;
          ;
               CREATE,0,.0004,,50,1;                    INITIAL A/C GENERATION FOR SQ5
                  ACT,,XX(54).EQ.1,NAL5;
                  ACT,,XX(54).EQ.0,TERM;
               CREATE,0,1800.0004,,50,1;                IF DAY02 REPLACEMENT SQ5 REQUIRE
                  ACT,,XX(54).EQ.1,NAL5;
                  ACT,,XX(54).EQ.0,TERM;
               CREATE,0,3240.0004,,50,1;                IF DAY03 REPLACEMENT SQ5 REQUIRE
                  ACT,,XX(54).EQ.1,NAL5;
                  ACT,,XX(54).EQ.0,TERM;
          ;
          NAL5 ASSIGN,XX(56)=XX(56)+1,1;
               ACT,,XX(56).GT.XX(57).AND.
                    TNOW.LT.0.1,TERM;
               ACT,,XX(56).GT.XX(57),RESC;
               ACT,,,ASQ5;
          ;                                             A/C IDENTIFICATION (SQ,TAIL #)
          ;                                             AND STATUS CODES.
          ASQ5 ASSIGN,XX(72)=XX(72)+1,
                    ATRIB(26)=0,ATRIB(27)=0,
                    ATRIB(1)=5, ATRIB(2)=USERF(18),
                    ATRIB(7)=USERF(137),
                    ATRIB(12)=XX(59),ATRIB(14)=0,ATRIB(15)=0,
                    ATRIB(16)=0,ATRIB(17)=0,ATRIB(18)=0;
          ;
          ;                                             A/C STATISTICS AND
          ;                                             NEXT TIME OF FAILURE, BY SYSTEM.
               ASSIGN,
                    ATRIB(4)=0, ATRIB(5)=0, ATRIB(6)=0,
```

```
                    ATRIB(28)=0,ATRIB(29)=0,ATRIB(30)=0,
                  ATRIB(19)=USERF(131),ATRIB(20)=USERF(132),
                  ATRIB(21)=USERF(133),ATRIB(22)=USERF(134),
                  ATRIB(23)=USERF(135),ATRIB(24)=USERF(136),1;
            ACT,,XX(54).EQ.1.AND.
                 TNOW.GT.0.1,RESC;        REPLACEMENT A/C
            ACT,,DRAND.GT.XX(64),MXIS;     A/C IN MAINT INITIALLY
            ACT,,,ASC5;                    GENERATED OPERATIONAL A/C
;
;                                          INITIALIZE CONFIGURATION
ASC5 ASSIGN,ATRIB(3)=USERF(21),
            ATRIB(9)=1, ATRIB(10)=1,ATRIB(11)=0,
            XX(78)=XX(78)+1,1;
            ACT,,XX(78).LE.XX(61),QRA;    SELECT QRA A/C (FIRST TIME ONLY)
            ACT,,XX(78).GT.XX(61);        OPERATIONAL AIRCRAFT
;
;
;
;
;                                          FORM A FLIGHT OF THREE
;
;                                          GATE OPENED BY SCHEDULER
;
ARP5 AWAIT(13),RDYPOOL5,1;                 READY POOL SQ5
            ACT,,ATRIB(12).NE.XX(48),GA15; IF WRONG CONFIG, RETURN TO RDYPOOL
            ACT,,ATRIB(12).EQ.XX(48),ACC5;
GA15 CLOSE,RDYPOOL5,1;
            ACT,,,ARP5;
;
ACC5 ASSIGN,XX(73)=XX(73)+1,
             ATRIB(43)=XX(49),ATRIB(44)=3,
             ATRIB(45)=XX(73),ATRIB(46)=XX(47),
            ATRIB(47)=XX(97),1;            ASSIGN MISSION INFO
            ACT,,XX(73).GT.3,ARP5;          RETURN REST TO READY POOL
            ACT,,XX(73).LE.3,GAT5;
;
GAT5 CLOSE,RDYPOOL5,1;
            ACT,,,ACS5;                    TO MATCH WITH PILOT
;
;
;
ACS5 QUEUE(15),,,,APM5;                    TO MATCH OF A/C & PILOTS
;
;
;
;
     CREATE,0,.0004,,75,1;                 INITIAL PILOT GENERATION FOR SQ5
            ACT,,XX(54).EQ.1,ASP5;
            ACT,,XX(54).EQ.0,TERM;
;
```

```
;                                       PILOT ID (SQ,ID#) AND
;                                       PILOT ACTIVITY STATISTICS.
 ASP5 ASSIGN,XX(70)=XX(70)+1,XX(71)=XX(71)+1,
            ATRIB(1)=5,ATRIB(2)=XX(70),
            ATRIB(3)=USERF(11),
            ATRIB(4)=0,ATRIB(5)=0,
            ATRIB(6)=0,ATRIB(11)=0,
            ATRIB(12)=0,
            ATRIB(8)=0,ATRIB(9)=0,1;
        ACT,,XX(71).EQ.75,QA5;
        ACT,,XX(71).LE.XX(61),PQRP;
        ACT,,XX(71).LE.XX(58),PL5;
        ACT,,,TERM;
;
 QA5  ASSIGN,XX(71)=0,XX(56)=0,1;              AUTO RESET OF COUNTER
        ACT,,XX(58).EQ.75,PL5;
        ACT,,,TERM;
;
;
 PL5  QUEUE(14),,,,APM5;                       TO A/C/PILOT MATCH
;
 APM5 MATCH,48,PL5/APLT,ACS5/ASAC;             ASSIGN A/C TO PILOTS
;
;
;
;
;
;                                       SQ6 A/C GENERATION ROUTINE
;
;
     CREATE,0,.0005,,50,1;                    INITIAL A/C GENERATION FOR SQ6
        ACT,,XX(55).EQ.1,NAL6;
        ACT,,XX(55).EQ.0,TERM;
     CREATE,0,1800.0005,,50,1;                IF DAY02 REPLACEMENT SQ6 REQUIRE
        ACT,,XX(55).EQ.1,NAL6;
        ACT,,XX(55).EQ.0,TERM;
     CREATE,0,3240.0005,,50,1;                IF DAY03 REPLACEMENT SQ6 REQUIRE
        ACT,,XX(55).EQ.1,NAL6;
        ACT,,XX(55).EQ.0,TERM;
;
 NAL6 ASSIGN,XX(56)=XX(56)+1,1;
        ACT,,XX(56).GT.XX(57).AND.
             TNOW.LT.0.1,TERM;
        ACT,,XX(56).GT.XX(57),RESC;
        ACT,,,ASQ6;
;                                       A/C IDENTIFICATION (SQ,TAIL #)
;                                       AND STATUS CODES.
 ASQ6 ASSIGN,XX(72)=XX(72)+1,
            ATRIB(26)=0,ATRIB(27)=0,
            ATRIB(1)=6, ATRIB(2)=USERF(18),
```

```
                    ATRIB(7)=USERF(137),
                ATRIB(12)=XX(59),ATRIB(14)=0,ATRIB(15)=0,
                ATRIB(16)=0,ATRIB(17)=0,ATRIB(18)=0;
;
;                                       A/C STATISTICS AND
;                                       NEXT TIME OF FAILURE, BY SYSTEM.
     ASSIGN,
            ATRIB(4)=0, ATRIB(5)=0, ATRIB(6)=0,
            ATRIB(28)=0,ATRIB(29)=0,ATRIB(30)=0,
            ATRIB(19)=USERF(131),ATRIB(20)=USERF(132),
            ATRIB(21)=USERF(133),ATRIB(22)=USERF(134),
            ATRIB(23)=USERF(135),ATRIB(24)=USERF(136),1;
          ACT,,XX(55).EQ.1.AND.
               TNOW.GT.0.1,RESC;        REPLACEMENT A/C
          ACT,,DRAND.GT.XX(64),MXIS;     A/C IN MAINT INITIALLY
          ACT,,,ASC6;                   GENERATED OPERATIONAL A/C
;
;                                       INITIALIZE CONFIGURATION
 ASC6 ASSIGN,ATRIB(3)=USERF(21),
            ATRIB(9)=1, ATRIB(10)=1,ATRIB(11)=0,
            XX(79)=XX(79)+1,1;
          ACT,,XX(79).LE.XX(61),QRA;    SELECT QRA A/C (FIRST TIME ONLY)
          ACT,,XX(79).GT.XX(61);        OPERATIONAL AIRCRAFT
;
;
;
;
;                                       FORM A FLIGHT OF THREE
;
;                                       GATE OPENED BY SCHEDULER
;
 ARP6 AWAIT(16),RDYPOOL6,1;             READY POOL SQ6
        ACT,,ATRIB(12).NE.XX(48),GA16; IF WRONG CONFIG, RETURN TO RDYPOOL
        ACT,,ATRIB(12).EQ.XX(48),ACC6;
 GA16 CLOSE,RDYPOOL6,1;
        ACT,,,ARP6;
;
 ACC6 ASSIGN,XX(73)=XX(73)+1,
         ATRIB(43)=XX(49),ATRIB(44)=3,
         ATRIB(45)=XX(73),ATRIB(46)=XX(47),
         ATRIB(47)=XX(97),1;            ASSIGN MISSION INFO
        ACT,,XX(73).GT.3,ARP6;          RETURN REST TO READY POOL
        ACT,,XX(73).LE.3,GAT6;
;
 GAT6 CLOSE,RDYPOOL6,1;
        ACT,,,ACS6;                     TO MATCH WITH PILOT
;
;
;
 ACS6 QUEUE(18),,,,APM6;                TO MATCH OF A/C & PILOTS
```

```
1013            ;
1014            ;
1015            ;
1016            ;
1017                 CREATE,0,.0005,,75,1;          INITIAL PILOT GENERATION FOR SQ6
1018                    ACT,,XX(55).EQ.1,ASP6;
1019                    ACT,,XX(55).EQ.0,TERM;
1020            ;
1021            ;                                   PILOT ID (SQ,ID#) AND
1022            ;                                   PILOT ACTIVITY STATISTICS.
1023             ASP6 ASSIGN,XX(70)=XX(70)+1,XX(71)=XX(71)+1,
1024                         ATRIB(1)=6,ATRIB(2)=XX(70),
1025                         ATRIB(3)=USERF(11),
1026                         ATRIB(4)=0,ATRIB(5)=0,
1027                         ATRIB(6)=0,ATRIB(11)=0,
1028                         ATRIB(12)=0,
1029                         ATRIB(8)=0,ATRIB(9)=0,1;
1030                    ACT,,XX(71).EQ.75,QA6;
1031                    ACT,,XX(71).LE.XX(61),PQRP;
1032                    ACT,,XX(71).LE.XX(58),PL6;
1033                    ACT,,,TERM;
1034            ;
1035             QA6  ASSIGN,XX(71)=0,XX(56)=0,1;          AUTO RESET OF COUNTER.
1036                    ACT,,XX(58).EQ.75,PL6;
1037                    ACT,,,TERM;
1038            ;
1039            ;
1040             PL6  QUEUE(17),,,,APM6;            TO A/C/PILOT MATCH
1041            ;
1042             APM6 MATCH,48,PL6/APLT,ACS6/ASAC;     ASSIGN A/C TO PILOTS
1043            ;
1044            ;
1045            ;
1046            ;                                   ROUTINE TO TRANSFER PILOT
1047            ;                                   ATTRIBUTES TO GLOBAL VARIABLES
1048             APLT ASSIGN,XX(80)=ATRIB(1), XX(81)=ATRIB(2),
1049                         XX(82)=ATRIB(3), XX(83)=ATRIB(4),
1050                         XX(84)=ATRIB(5), XX(85)=ATRIB(6),
1051                         XX(86)=ATRIB(7), XX(87)=ATRIB(8),
1052                         XX(88)=ATRIB(9), XX(89)=ATRIB(10),
1053                         XX(90)=ATRIB(11),XX(91)=ATRIB(12);
1054                   TERMINATE;
1055            ;
1056            ;
1057            ;                                   ROUTINE TO TRANSFER PILOT
1058            ;                                   ATTRIBUTES FROM GLOBAL
1059            ;                                   VARIABLES TO A/C
1060             ASAC ASSIGN,ATRIB(31)=XX(80),ATRIB(32)=XX(81),
1061                         ATRIB(33)=XX(82),ATRIB(34)=XX(83),
1062                         ATRIB(35)=XX(84),ATRIB(36)=XX(85),
```

```
1063                    ATRIB(37)=XX(86),ATRIB(38)=XX(87),
1064                  ATRIB(39)=XX(88),ATRIB(40)=XX(89);
1065          ;
1066               ASSIGN,ATRIB(41)=XX(90),
1067                      ATRIB(42)=XX(91),1;
1068          ;
1069          ;
1070          ;                                    ROUTINE TO COLLECT STATISTICS
1071          ;                                    ON THE AMOUNT OF TIME A PILOT
1072          ;                                    HAS ON THE GROUND BETWEEN
1073          ;                                    FLIGHTS. TO VALIDATE FLIGHT
1074          ;                                    PLANNING AND BRIEFING TIME
1075          ;                                    AVAILABILITY.
1076                  ACT,,TNOW.GE.1440.0.AND.
1077                       ATRIB(41).EQ.0.AND.
1078                       ATRIB(42).EQ.0.OR.
1079                       TNOW.GE.2880.0.AND.
1080                       ATRIB(42).EQ.0,PPFT;    BRANCH AROUND STAT ON 1ST MSN
1081          ;                                    OF DAY ELSE GO TO STAT NODE
1082          ;                                    FOR TIME ON GROUND FOR PILOT
1083          ;                                    BETWEEN FLIGHTS.
1084                  ACT,,ATRIB(42).NE.0,CL06;
1085                  ACT,,ATRIB(41).NE.0,CL05;
1086                  ACT,,ATRIB(40).NE.0,CL04;
1087                  ACT,,,PPFT;                 BRANCH FOR FIRST TIME THRU,DAY01
1088          ;
1089           CL06 COLCT,INTVL(42),PILOTGRNDTIME03,,1;  DAY 03
1090                  ACT,,,PPFT;
1091           CL05 COLCT,INTVL(41),PILOTGRNDTIME02,,1;  DAY 02
1092                  ACT,,,PPFT;
1093           CL04 COLCT,INTVL(40),PILOTGRNDTIME01,,1;  DAY 01
1094                  ACT,,,PPFT;
1095          ;
```

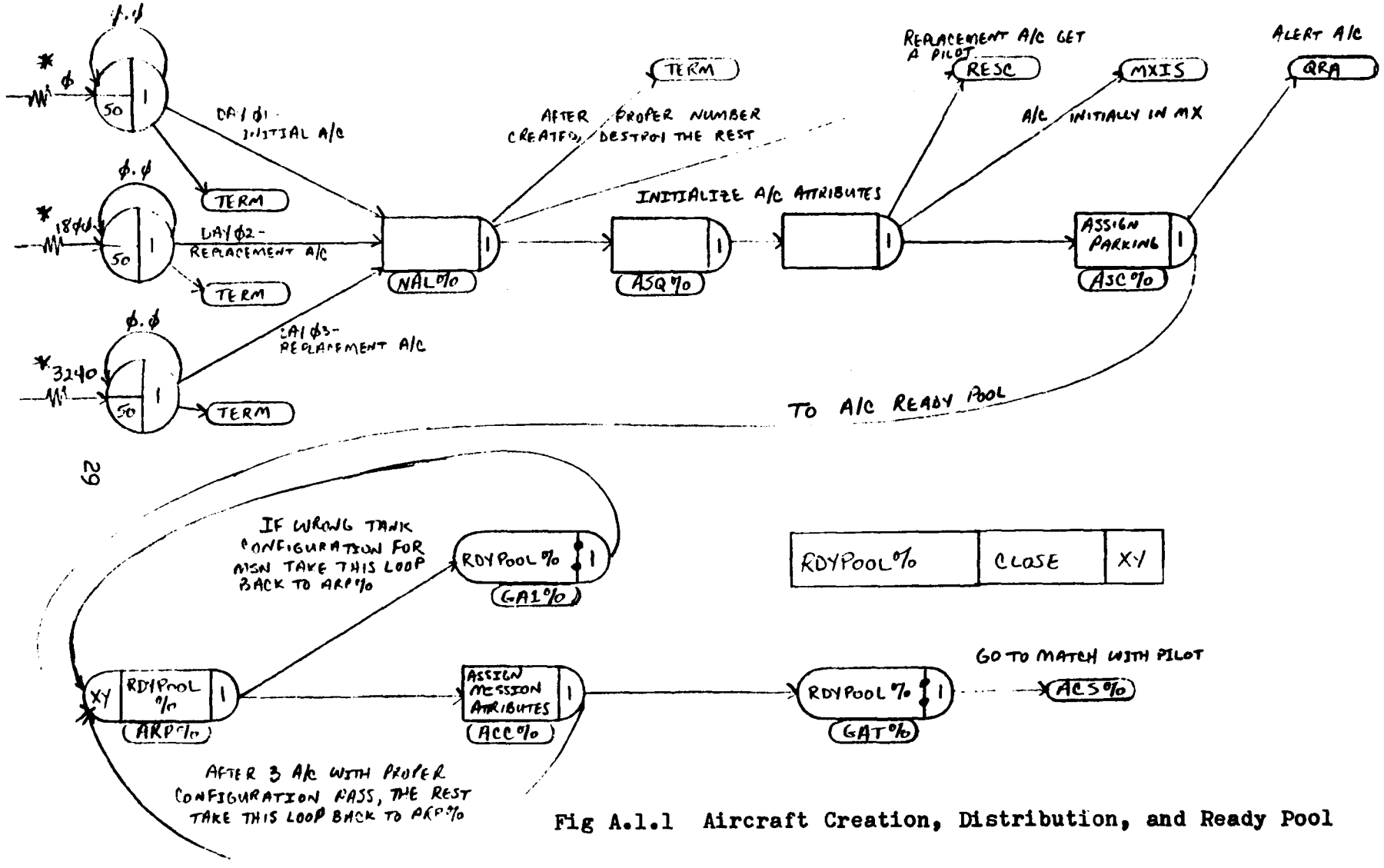
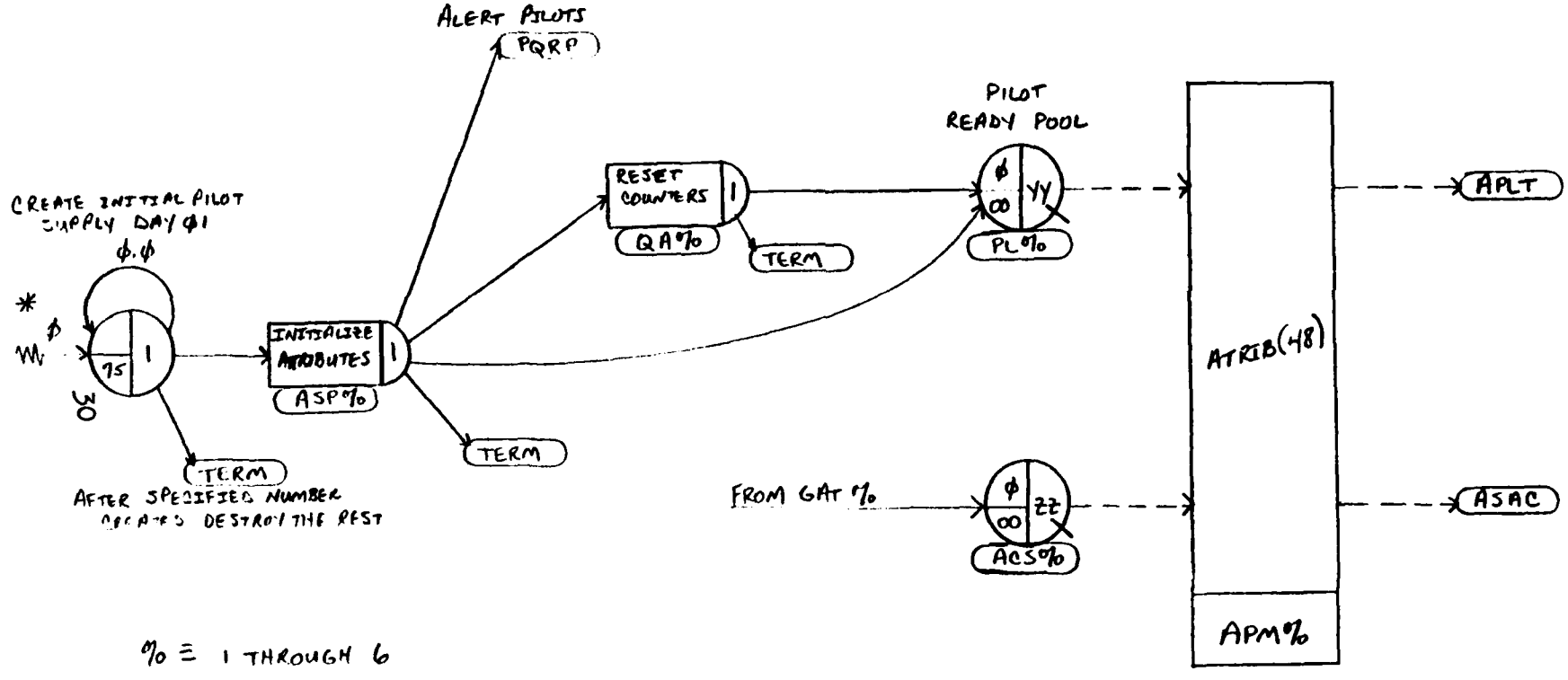


Fig A.1.1 Aircraft Creation, Distribution, and Ready Pool

XY ≡ (1, 4, 7, 10, 13, 16)

% ≡ 1 THROUGH 6

* SUBSEQUENT SQDN. TIMES HAVE .0φφ1 ADDED
  E.G. SQ 2 φ.φφφ1, 1800.φφφ1, 3240.φφφ1
  SQ 3 φ.φφφ2, 1800.φφφ2, 3240.φφφ2



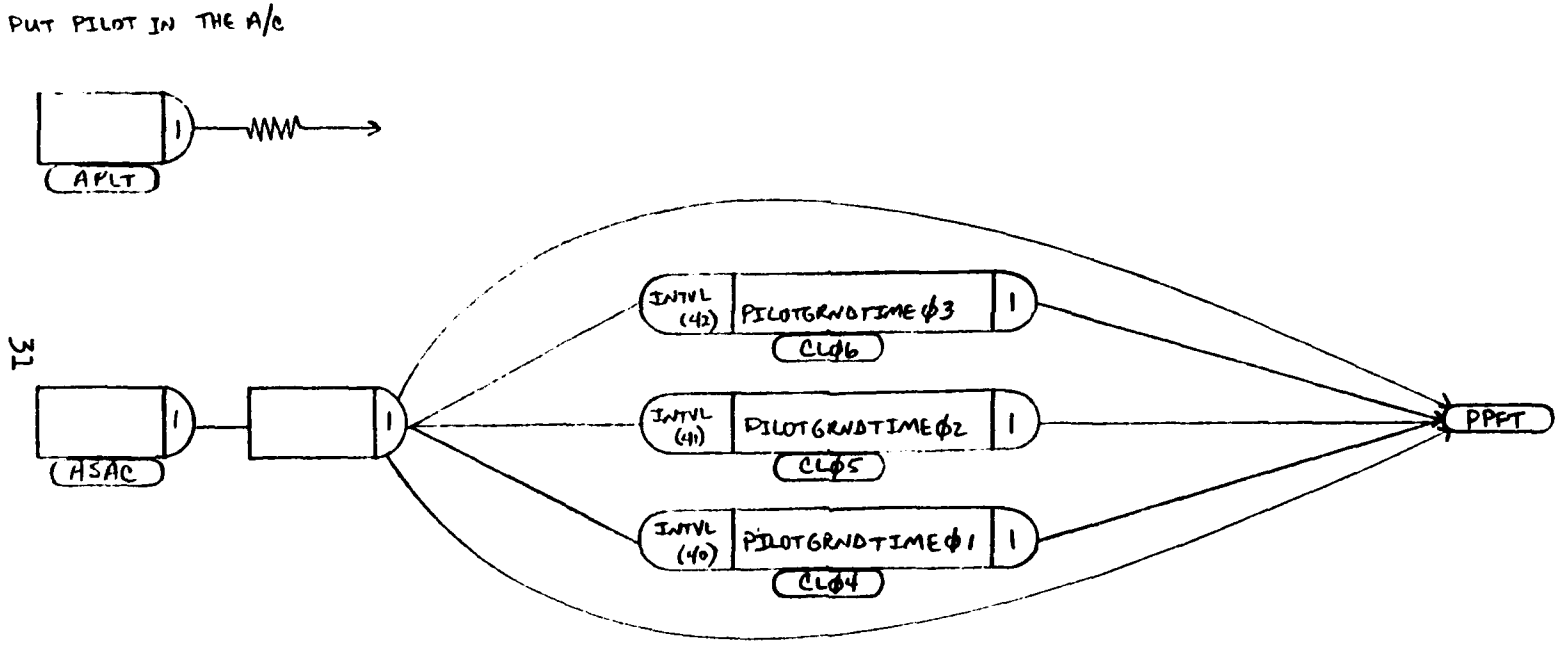
% ≡ 1 THROUGH 6

YY ≡ (2, 5, 8, 11, 14, 17)

ZZ ≡ (3, 6, 9, 12, 15, 18)

* SUBSEQUENT SQDN. TIMES HAVE .ϕϕϕ1 ADDED

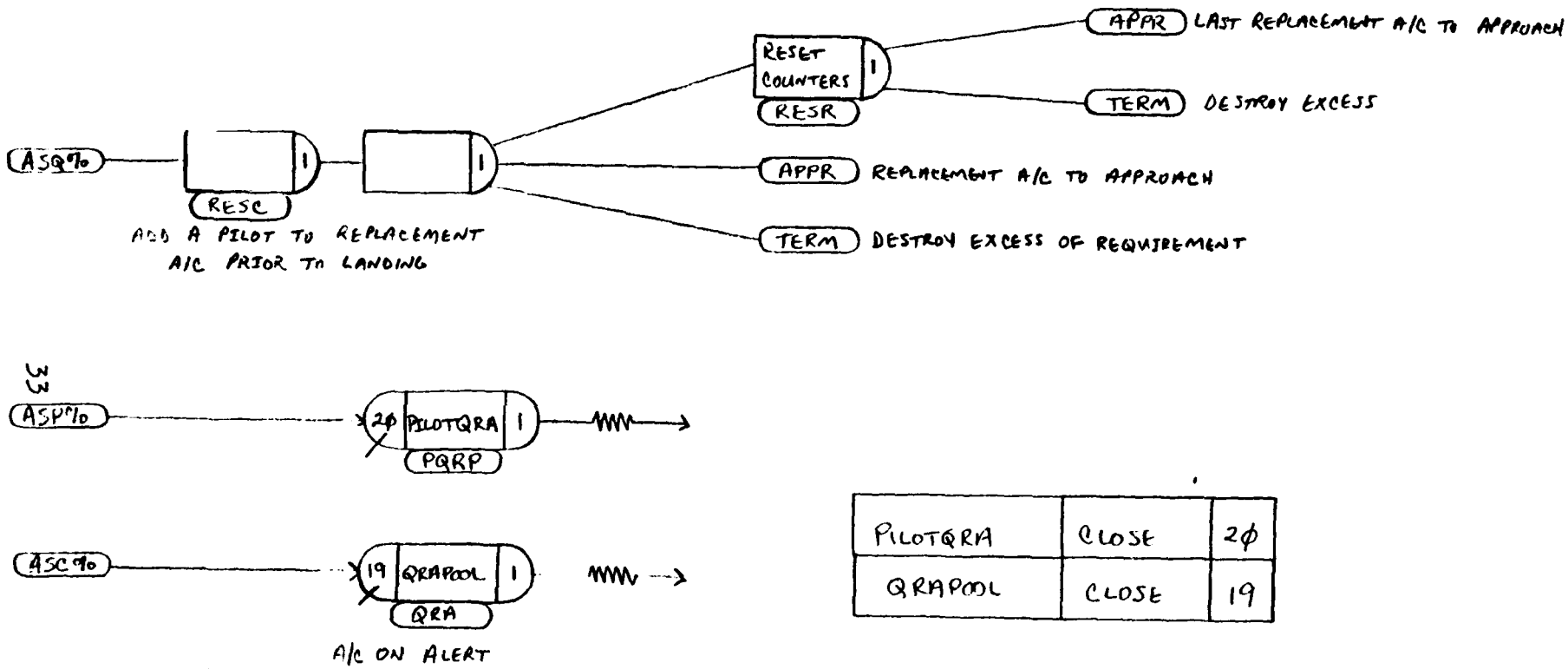
Fig A.1.2 Pilot Creation and Aircraft/Pilot Match



STATISTICS ON THE TIME BETWEEN ENGINE SHUTDOWN AND NEXT PILOT PREFLIGHT

Fig A.1.3 Creation Common Area

```
 1096          ; REPLACEMENT A/C AND PILOT INITIALIZATION AND QRA
1097          ;      WHEN RESUPPLY IS SCHEDULED, AT THE DESIGNATED TIME THE A/C
1098          ; ARE CREATED AND ROUTED TO RESC WHERE THE PILOT IS INITIALIZED AND
1099          ; PLACED IN THE A/C.  THE AIRCRAFT THEN PROCEEDS TO APPROACH TO
1100          ; OBTAIN THE RUNWAY AND LAND.
1101          ; BECAUSE THE REPLACEMENT PILOTS ARE CREATED ON A ONE FOR ONE BASIS
1102          ; WITH REPLACEMENT A/C THE USER PROVIDED NUMBER OF PILOTS PER SQUAD-
1103          ; RON DOES NOT HOLD TRUE AFTER RESUPPLY.  TO ATTEMPT TO CONFORM TO
1104          ; THE USERS INTENT, THE RATIO OF PILOT QUALIFICATION STATUS (FLIGHT
1105          ; LEAD, ETC.) IS MAINTAINED.
1106          ;     AIRCRAFT ASSIGNED TO QRA ARE STORED IN THE AIRCRAFT QRA AWAIT NODE
1107          ; PILOTS SELECTED FOR QRA ARE PLACED IN THE PILOT QRA NODE FILE.
1108          ;
1109          ;                                    ROUTINE TO SET UP ATTRIBUTES OF
1110          ;                                      PILOTS LANDING WITH A REPLACE-
1111          ;                                      MENT SQUADRON
1112           RESC ASSIGN,XX(70)=XX(70)+1,
1113                       XX(71)=XX(71)+1,ATRIB(31)=ATRIB(1),
1114                       ATRIB(32)=XX(70),ATRIB(33)=USERF(11),
1115                       ATRIB(34)=0,ATRIB(35)=0,ATRIB(36)=0,
1116                       ATRIB(38)=0,ATRIB(39)=0,1;
1117          ;
1118                ASSIGN,
1119                    ATRIB(41)=0,ATRIB(42)=0,
1120                    ATRIB(8)=TNOW-50,1;
1121                   ACT,,XX(71).EQ.50,RESR;
1122                   ACT,,XX(71).LE.XX(57),APPR;
1123                  ACT,,,TERM;                  REPLACEMENT SQ A/C TO TERM
1124          ;
1125           RESR ASSIGN,XX(71)=0,XX(56)=0,1;
1126                  ACT,,XX(57).EQ.50,APPR;
1127                  ACT,,,TERM;
1128          ;
1129          ;
1130           PQRP AWAIT(20),PILOTQRA,1;          QRA PILOT HOLDING AREA
1131                TERMINATE;
1132          ;
1133           QRA  AWAIT(19),QRAPOOL,1;           QRA A/C HOLDING AREA
1134                TERMINATE;
1135          ;
```



33

PILOTQRA	CLOSE	2ϕ
QRAPOOL	CLOSE	19

Fig A.2.1 Replacement Squadron Processing and Quick Reaction Alert (QRA)

% = 1 THROUGH 6

```
1136            ; INITIALIZATION OF AIRCRAFT IN MAINTENANCE
1137           ; A USER SELECTABLE PERCENTAGE OF AIRCRAFT ARE NOT INITIALLY OR.
1138           ; THESE AIRCRAFT ARE DISTRIBUTED AMONG THE MAINTENANCE SERVICE
1139           ; FACILITIES IN A UNIFORM MANNER.  AIRCRAFT WITH LEVEL 4 OR 5
1140           ; PROBLEMS ARE REPAIRED AT WING OR BY AN MMT, WHILE 2 AND 3 LEVEL
1141           ; PROBLEMS ARE REPAIRED AT SQUADRON.
1142           ;
1143            MXIS ASSIGN,ATRIB(9)=0,ATRIB(10)=0,
1144                    ATRIB(11)=0,ATRIB(18)=USERF(16),
1145                    ATRIB(3)=USERF(23),1;        ASSIGN SQ PARKING TO MX A/C
1146                   ACT,,USERF(37).GE.4,MXRT;     MAJOR MAINTENANCE ROUTE (WG/MMT)
1147                   ACT,,USERF(37).LT.4,SPMX;     TO SQ MX
1148           ;
1149            MXRT GOON,1;                         DISTRIBUTE INITIALLY BROKEN A/C
1150                   ACT,,.66,WGPR;                TO WING
1151                   ACT,,.34,SPMX;                TO SPMX
1152           ;
1153            WGPR ASSIGN,ATRIB(3)=USERF(22),
1154                       ATRIB(3)=4,1;             UNPARK AT SQ, PARK AT WING
1155                   ACT,,WG;
```

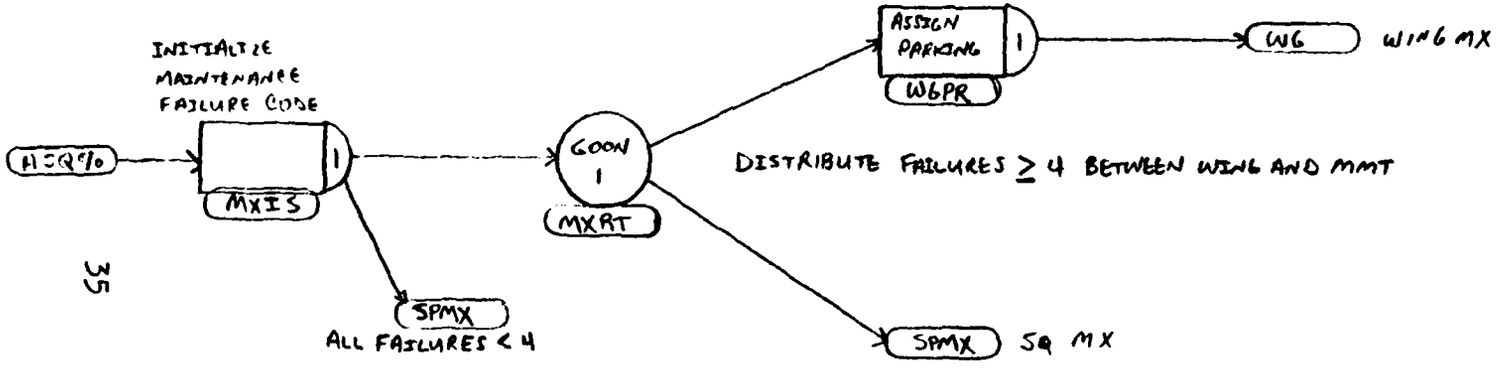


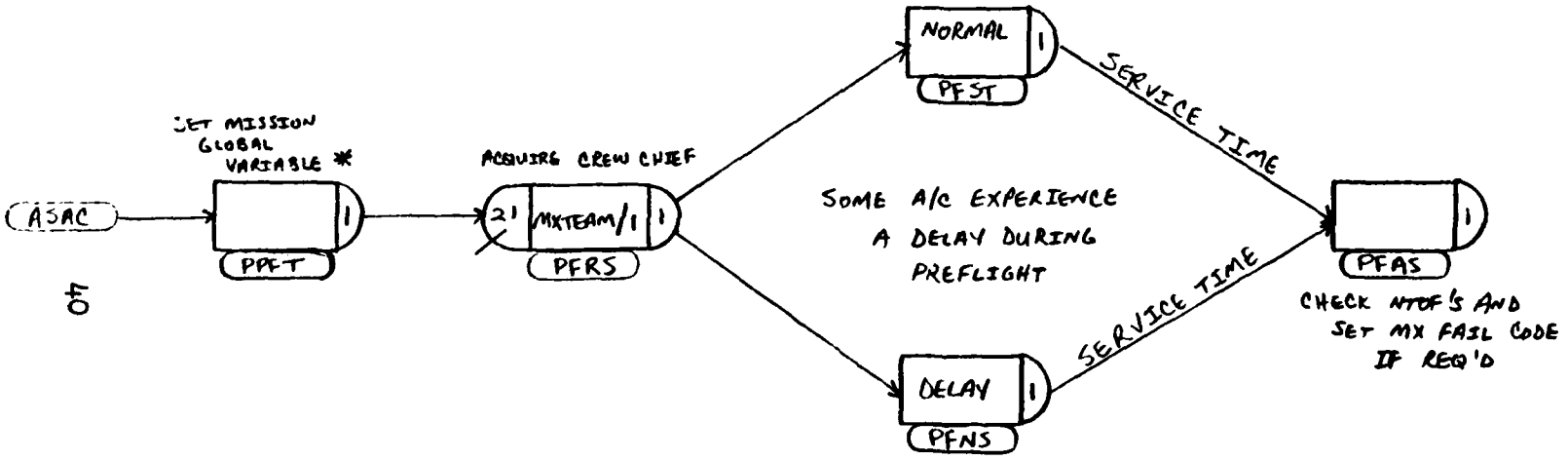
Fig A.3.1 Initialization of Aircraft in Maintenance

```
1156            ; PILOT PREFLIGHT
1157           ;    AIRCRAFT ARRIVING AT PILOT PREFLIGHT ACQUIRE A CREW CHIEF AND RE-
1158           ; CEIVE NORMAL OR DELAYED SERVICE DUE TO A PROBLEM.  AIRCRAFT WITH
1159           ; DELAYED SERVICE HAVE A SLIGHTLY HIGHER PROBABILITY OF FAILURE
1160           ; BECAUSE A FAILURE WILL BE SHOWN IF THE AIRCRAFT IS WITHIN 5 MIN-
1161           ; UTES OF AN NTOF ON ANY SYSTEM.
1162           ;    IF THE LEAD AIRCRAFT IS BROKEN AND A SPARE IS NOT AVAILABLE, THE
1163           ; LEAD PILOT TAKES THE NUMBER THREE AIRCRAFT UNLESS THE NUMBER THREE
1164           ; PILOT IS FLIGHT LEAD QUALIFIED.
1165           ;    ONCE THE PREFLIGHT ACTIVITY IS FINISHED THE AIRCRAFT ARE EVALU-
1166           ; ATED FOR THEIR FAILURE STATUS.  BROKEN AIRCRAFT ATTEMPT TO GET
1167           ; A SPARE. THE FLIGHT THEN PROCEEDS AS A THREE-SHIP, A TWO-SHIP,
1168           ; OR THE MISSION IS SCRUBBED.
1169           ;
1170            PPFT ASSIGN,II=ATRIB(46),XX(II)=0,1;  CLEAR MISSION (MSN) STATUS CODE
1171                   ACT,,,PFRS;
1172           ;
1173            PFRS AWAIT(21),MXTEAM/1,1;            WAIT FOR A CREW CHIEF
1174                   ACT,,DRAND.GT.XX(65),PFST;     XX(65)=% TIME PREFLT DELAYED
1175                   ACT,,,PFNS;
1176           ;
1177            PFST ASSIGN,ATRIB(8)=TRIAG(4,5,7),1;  NORMAL PILOT PRE-FLIGHT
1178                   ACT/41,ATRIB(8),,PFAS;
1179           ;
1180            PFNS ASSIGN,ATRIB(18)=USERF(52),
1181                        ATRIB(8)=TRIAG(4,5,7)+
1182                                 TRIAG(6,10,12),1;PILOT PRE-FLIGHT W/DELAY
1183                   ACT/42,ATRIB(8),,PFAS;
1184           ;
1185            PFAS ASSIGN,ATRIB(18)=USERF(51),1;    UPDATE DYNAMIC FAILURE CODE
1186                   ACT,,ATRIB(45).EQ.1.AND.
1187                        USERF(37).GE.2,PEV1;      A/C 1 FAIL
1188                   ACT,,ATRIB(45).EQ.1.AND.
1189                        USERF(37).LT.2,QPF1;      A/C 1 OK
1190                   ACT,,ATRIB(45).EQ.2.AND.
1191                        USERF(37).GE.2,PEV2;      A/C 2 FAIL
1192                   ACT,,ATRIB(45).EQ.2.AND.
1193                        USERF(37).LT.2,QPF2;      A/C 2 OK
1194                   ACT,,ATRIB(45).EQ.3.AND.
1195                        USERF(37).GE.2,PEV3;      A/C 3 FAIL
1196                   ACT,,ATRIB(45).EQ.3.AND.
1197                        USERF(37).LT.2,QPF3;      A/C 3 OK
1198           ;
1199            PEV1 EVENT,10,1;                      GET SPARE A/C IF AVAILABLE
1200                   ACT,,XX(95).EQ.1,PFA1;
1201                   ACT,,XX(95).EQ.0,PFF1;
1202           ;
1203            PFF1 ASSIGN,II=ATRIB(46),
1204                   XX(II)=XX(II)+2,1;             SET MISSION (MSN) STATUS CODE
1205                   ACT,,,QPF1;
```

```
1206            ;
1207            QPF1 QUEUE(22),,,,PFMA;              QUEUE TO MATCH A/C 1 W/MSN#
1208            ;
1209            PEV2 EVENT,10,1;                     GET SPARE A/C IF AVAILABLE
1210                    ACT,,XX(95).EQ.1,PFA1;
1211                    ACT,,XX(95).EQ.0,PFF2;
1212            ;
1213            PFF2 ASSIGN,II=ATRIB(46),
1214                    XX(II)=XX(II)+4,1;           SET MSN STATUS CODE
1215                    ACT,,,QPF2;
1216            ;
1217            QPF2 QUEUE(23),,,,PFMA;              QUEUE TO MATCH A/C 2 W/MSN#
1218            ;
1219            PEV3 EVENT,10,1;                     GET SPARE A/C IF AVAILABLE
1220                    ACT,,XX(95).EQ.1,PFA1;
1221                    ACT,,XX(95).EQ.0,PFF3;
1222            ;
1223            PFF3 ASSIGN,II=ATRIB(46),
1224                    XX(II)=XX(II)+5,1;           SET MSN STATUS CODE
1225                    ACT,,,QPF3;
1226            ;
1227            QPF3 QUEUE(24),,,,PFMA;              QUEUE TO MATCH A/C 3 W/MSN#
1228            ;
1229            PFMA MATCH,46,QPF1/PFS1,QPF2/PFT1,
1230                    QPF3/PFU1;                   MATCH A/C BY MSN#
1231            ;
1232            PFS1 ASSIGN,II=ATRIB(46),1;
1233                    ACT,,XX(II).EQ.2.OR.XX(II).EQ.6.OR.
1234                         XX(II).EQ.7.OR.
1235                         XX(II).EQ.11,PFS2;      IF A/C 1 FAIL
1236                    ACT,,XX(II).EQ.9,STRP;       IF A/C 1 SYM ABORT
1237                    ACT,,XX(II).EQ.4.OR.
1238                      XX(II).EQ.5,PFA2;          IF A/C 1 NOW IN TWO SHIP
1239                    ACT,,XX(II).EQ.0,ST3;        IF A/C 1 STILL IN THREE SHIP
1240            ;
1241            PFS2 COON,2;
1242                    ACT,,ATRIB(43).NE.1,PSEP;
1243                    ACT,,ATRIB(43).NE.1,PFA1;    IF NOT IN CASE 1
1244                    ACT,,ATRIB(43).EQ.1,PFS3;    CASE 1
1245            ;
1246            ;                                    TRANSFER PILOT 1 ATTRIBUTES
1247            ;                                    TO GLOBAL VARIABLES
1248            PFS3 ASSIGN,XX(81)=ATRIB(32),XX(82)=ATRIB(33),
1249                        XX(83)=ATRIB(34),XX(84)=ATRIB(35),
1250                        XX(85)=ATRIB(36),XX(86)=ATRIB(37),
1251                        XX(87)=ATRIB(38),XX(88)=ATRIB(39),
1252                        XX(89)=ATRIB(40),XX(90)=ATRIB(41),
1253                        XX(91)=ATRIB(42),ATRIB(32)=0,1;
1254                    ACT,,,PFA1;;
1255            ;
```

```
1256            PFA1 ASSIGN,ATRIB(32)=0,
1257                      ATRIB(13)=ATRIB(1),1;     ASSIGN A/C LOCATION CODE (BY SQ)
1258                 ACT,,,SMXC;
1259            ;
1260            ;
1261             PFT1 ASSIGN,II=ATRIB(46),2;
1262                    ACT,,XX(II).EQ.4.OR.XX(II).EQ.6.OR.
1263                         XX(II).EQ.9.OR.
1264                         XX(II).EQ.11,PSEP;       PILOT 2 RETURN TO READY POOL
1265                    ACT,,XX(II).EQ.4.OR.XX(II).EQ.6.OR.
1266                         XX(II).EQ.9.OR.
1267                         XX(II).EQ.11,PFA1;       A/C 2 BROKEN
1268                    ACT,,XX(II).EQ.7,STRP;        A/C 2 SYM ABORT
1269                    ACT,,XX(II).EQ.2.OR.
1270                       XX(II).EQ.5,PFA2;          A/C 2 NOW IN TWO SHIP
1271                    ACT,,XX(II).EQ.0,ST3;         A/C 2 STILL IN THREE SHIP
1272            ;
1273             PFU1 ASSIGN,II=ATRIB(46),2;
1274                    ACT,,XX(II).EQ.5.OR.XX(II).EQ.7.OR.
1275                         XX(II).EQ.9.OR.
1276                         XX(II).EQ.11,PFU2;       A/C 3 BROKEN
1277                    ACT,,XX(II).EQ.6,STRP;        A/C 3 SYM ABORT
1278                    ACT,,XX(II).EQ.0,ST3;         A/C 3 STILL IN THREE SHIP
1279                    ACT,,XX(II).EQ.2.AND.
1280                       ATRIB(43).NE.1,PFU5;       A/C 3 NOW LEAD IN A TWO SHIP
1281                    ACT,,XX(II).EQ.4,PFU6;        A/C 3 NOW #2 IN A TWO SHIP
1282                    ACT,,XX(II).EQ.2.AND.
1283                       ATRIB(43).EQ.1,PSEP;       PILOT #3 RETURN TO READYPOOL
1284                    ACT,,XX(II).EQ.2.AND.
1285                       ATRIB(43).EQ.1,PFU7;       PILOT #1 TAKES A/C 3
1286            ;
1287             PFU2 ASSIGN,XX(II)=0,2;              BROKEN
1288                    ACT,,,PSEP;
1289                    ACT,,,PFA1;                   TO MX
1290            ;
1291             PFU5 ASSIGN,ATRIB(45)=1,XX(II)=0,1;  REASSIGNS A/C 3 AS A/C 1
1292                    ACT,,,PFA2;                   TWO SHIP
1293            ;
1294             PFU6 ASSIGN,ATRIB(45)=2,XX(II)=0,1;  ASSIGNS A/C 3 AS A/C 2
1295                    ACT,,,PFA2;                   TWO SHIP
1296            ;
1297            ;                                     PLACE PILOT 1 IN A/C 3
1298             PFU7 ASSIGN,ATRIB(32)=XX(81),ATRIB(33)=XX(82),
1299                       ATRIB(34)=XX(83),ATRIB(35)=XX(84),
1300                       ATRIB(36)=XX(85),ATRIB(37)=XX(86),
1301                       ATRIB(38)=XX(87),ATRIB(39)=XX(88),
1302                       ATRIB(40)=XX(89),ATRIB(41)=XX(90),
1303                       ATRIB(42)=XX(91),1;
1304                    ACT,,,PFU5;
1305            ;
```

```
 1306          PFA2 ASSIGN,ATRIB(44)=2,1;          SETS CODE FOR TWO SHIP
1307               ACT,,,ST2;                     TO START TWO
1308          ;
```



* MISSION GLOBAL VARIABLE (MGV) – INDICATES CURRENT MAINTENANCE STATUS OF EACH A/C IN A FLIGHT

MXTEAM (96)	21	51	98

Fig A.4.1 Pilot Preflight

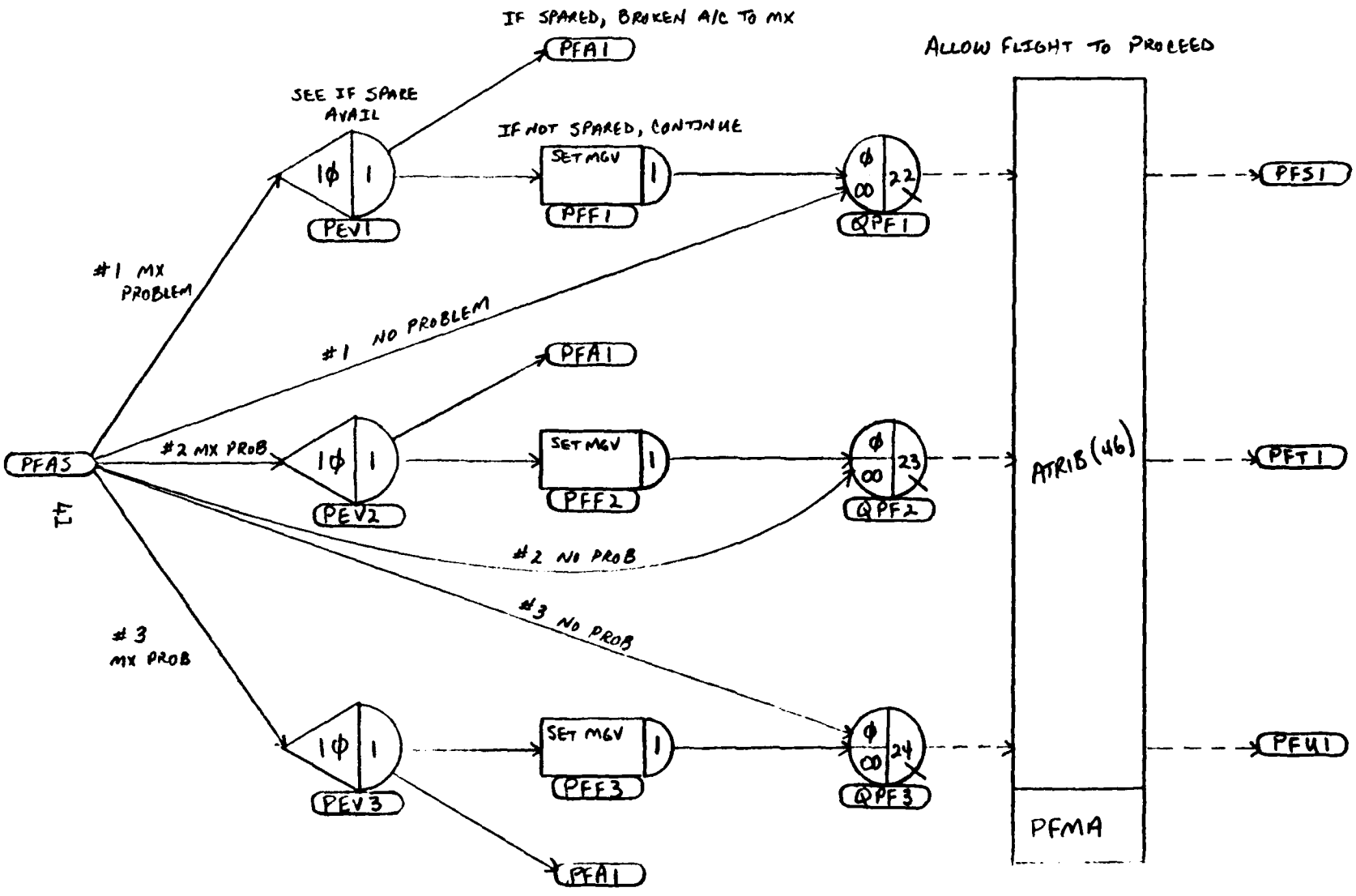


Fig A.4.2 Pilot Preflight

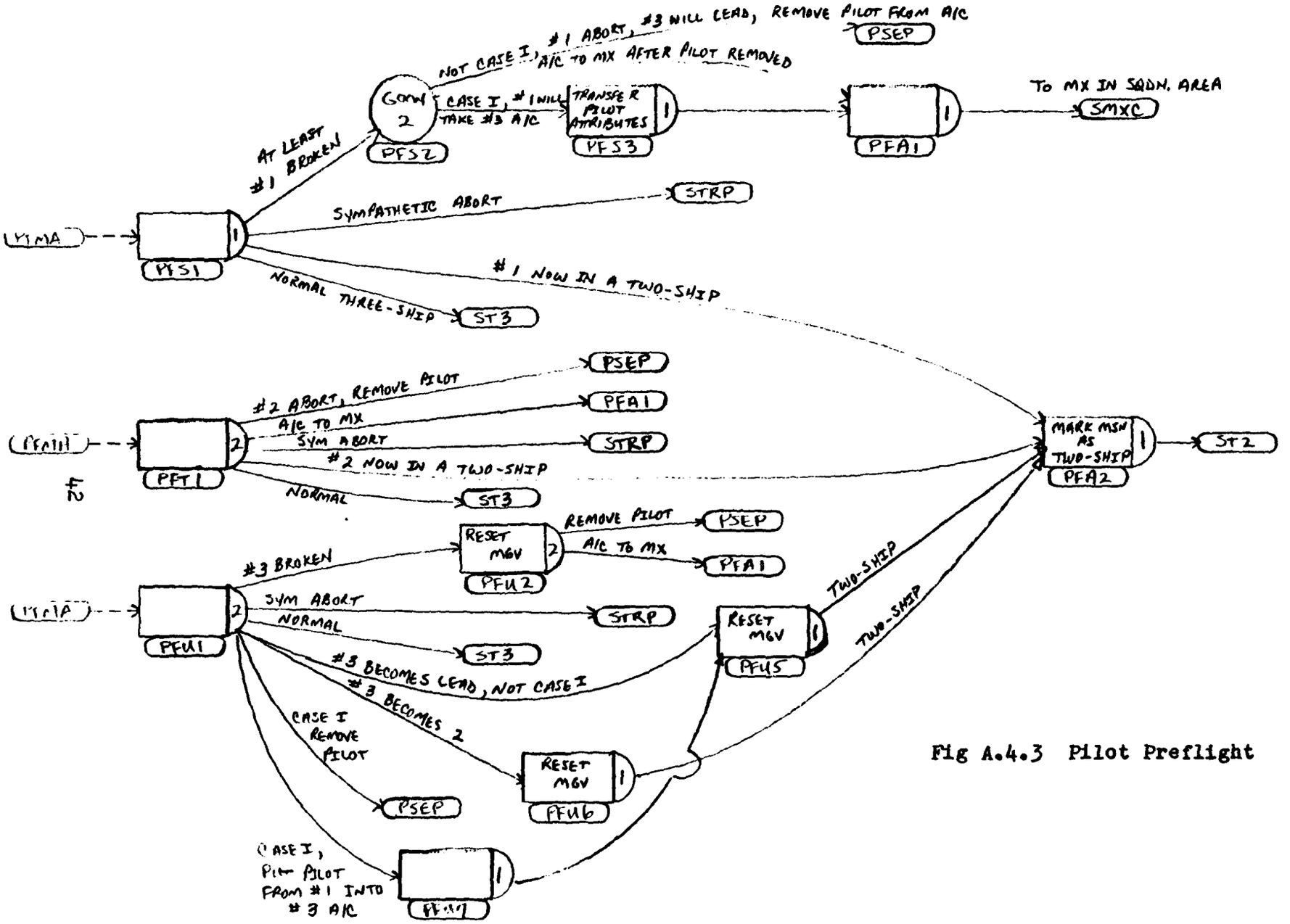


Fig A.4.3 Pilot Preflight

```
1309            ; THREE-SHIP START
1310           ;    THIS SECTION PROVIDES TIME FOR STRAP-IN AND PRE-START CHECKS.
1311           ; A/C THEN START ENGINES.  SOME MAY BE DELAYED ON START.  AFTER
1312           ; START, IN THE CHOCK CHECKS ARE ACCOMPLISHED.  ONCE THE A/C HAVE
1313           ; STARTED, ENGINE RUN TIME IS TRACKED.  THIS VALUE IS USED TO COM-
1314           ; PARE TO NTOF TO DETERMINE WHEN AN AIRCRAFT FAILS FOR A GIVEN
1315           ; SYSTEM.  ONCE CHECKS ARE COMPLETE, AIRCRAFT ARE EVALUATED FOR
1316           ; FAILURES AND ROUTED ACCORDINGLY.  AS WITH PREFLIGHT, IF THE LEAD
1317           ; A/C BREAKS THE PILOT TAKES THE NUMBER THREE A/C UNLESS ITS
1318           ; PILOT IS A FLIGHT LEAD.
1319           ;
1320            ST3  GOON,1;                          XX(66)- % OF A/C DELAYED STARTING
1321                   ACT,TRIAG(2,3,4),
1322                       DRAND.CT.XX(66),S3AS;      NORMAL PROCESS
1323                   ACT,TRIAG(4,5,6),,S3AS;        DELAYED PROCESS
1324           ;
1325            S3AS ASSIGN,ATRIB(8)=TNOW,1;          A/C OPERATING TIME HAS BEGUN
1326                   ACT,TRIAG(2,3,4),
1327                       ATRIB(45).EQ.1,QST1;       CONTROL SURFACE TESTING
1328                   ACT,TRIAG(2,3,4),
1329                       ATRIB(45).EQ.2,QST2;
1330                   ACT,TRIAG(2,3,4),
1331                       ATRIB(45).EQ.3,QST3;
1332           ;
1333            QST1 QUEUE(25),,,,S3MA;               WAIT TILL ALL 3 A/C ARE DONE
1334           ;
1335            QST2 QUEUE(26),,,,S3MA;
1336           ;
1337            QST3 QUEUE(27),,,,S3MA;
1338           ;
1339            S3MA MATCH,46,QST1/S3F1,QST2/S3F2,
1340                         QST3/S3F3;               REFORM FLIGHT
1341           ;
1342           ;                                     UPDATE MISSION STATUS CODE BY A/C
1343            S3F1 ASSIGN,ATRIB(8) =TNOW - ATRIB(8),
1344                       ATRIB(14)=ATRIB(14)+ATRIB(8),
1345                       ATRIB(7) =ATRIB(7) +ATRIB(8),
1346                       ATRIB(18)=USERF(51),
1347                       ATRIB(8) =TNOW,1;
1348                   ACT,,USERF(37).GE.2,S3F4;      A/C 1 BROKEN
1349                   ACT,.0001,,STS1;               A/C 1 OK
1350           ;
1351            S3F4 ASSIGN,II=ATRIB(46),
1352                       XX(II)=XX(II)+2,1;         SET UP TO GO TO MX
1353                   ACT,.0001,,STS1;
1354           ;
1355           ;                                     UPDATE MISSION STATUS CODE BY A/C
1356           ;
1357            S3F2 ASSIGN,ATRIB(8) =TNOW - ATRIB(8),
1358                       ATRIB(14)=ATRIB(14)+ATRIB(8),
```

```
                    ATRIB(7) =ATRIB(7) +ATRIB(8),
                    ATRIB(18)=USERF(51),
                    ATRIB(8) =TNOW,1;
               ACT,,USERF(37).GE.2,S3F5;     A/C 2 BROKEN
               ACT,.0001,,STT1;              A/C 2 OK
          ;
          S3F5 ASSIGN,II=ATRIB(46),
                    XX(II)=XX(II)+4,1;        SET UP TO GO TO MX
               ACT,.0001,,STT1;
          ;
          ;                                  UPDATE MISSION STATUS CODE BY A/C
          ;
          S3F3 ASSIGN,ATRIB(8) =TNOW - ATRIB(8),
                    ATRIB(14)=ATRIB(14)+ATRIB(8),
                    ATRIB(7) =ATRIB(7) +ATRIB(8),
                    ATRIB(18)=USERF(51),
                    ATRIB(8) =TNOW,1;
               ACT,,USERF(37).GE.2,S3F6;     A/C 3 BROKEN
               ACT,.0001,,STU1;              A/C 3 OK
          ;
          S3F6 ASSIGN,II=ATRIB(46),
                    XX(II)=XX(II)+5,1;        SET UP TO GO TO MX
               ACT,.0001,,STU1;
          ;
          ;
          ;                                  CHECK THE MISSION STATUS CODE TO
          ;                                  DETERMINE WHAT HAPPENED TO EACH
          ;                                  A/C IN THE FLIGHT
          STS1 ASSIGN,II=ATRIB(46),1;
               ACT,,XX(II).EQ.2.OR.XX(II).EQ.6.OR.
                    XX(II).EQ.7.OR.
                    XX(II).EQ.11,STS2;       A/C 1 FAILED
               ACT,,XX(II).EQ.9,STRP;        SYM ABORT
               ACT,,XX(II).EQ.4.OR.
                    XX(II).EQ.5,STA2;        A/C 1 PART OF TWO SHIP
               ACT,,XX(II).EQ.0,T3UP;        FLIGHT STILL A THREE SHIP
          ;
          STS2 ASSIGN,XX(95)=USERF(124),2;
               ACT,,ATRIB(43).NE.1,PSEP;
               ACT,,ATRIB(43).NE.1,S2LO;     A/C 1 FAIL, NOT CASE 1
               ACT,,ATRIB(43).EQ.1,STS3;     A/C 1 FAIL, CASE 1
          ;
          ;                                  PLACE PILOT 1 ATRIBUTES IN GLOBAL
          ;                                  VARIABLES SO THEY CAN BE PLACED
          ;                                  IN A/C 3.  ALSO, ZERO OUT PILOT
          ;                                  BECAUSE A/C 1 NO LONGER HAS A
          ;                                  PILOT IN IT.
          STS3 ASSIGN,XX(81)=ATRIB(32),XX(82)=ATRIB(33),
                    XX(83)=ATRIB(34),XX(84)=ATRIB(35),
                    XX(85)=ATRIB(36),XX(86)=ATRIB(37),
```

```
1409                    XX(87)=ATRIB(38),XX(88)=ATRIB(39),
1410                  XX(89)=ATRIB(40),XX(90)=ATRIB(41),
1411                  XX(91)=ATRIB(42),ATRIB(32)=0,1;
1412            ACT,,,S2LO;
1413        ;
1414         STT1 ASSIGN,II=ATRIB(46),
1415                    XX(95)=USERF(124),2;
1416            ACT,,XX(II).EQ.4.OR.XX(II).EQ.6.OR.
1417                 XX(II).EQ.9.OR.
1418                 XX(II).EQ.11,PSEP;       PILOT SEPERATION
1419            ACT,,XX(II).EQ.4.OR.XX(II).EQ.6.OR.
1420                 XX(II).EQ.9.OR.
1421                 XX(II).EQ.11,S2LO;       A/C 2 FAILED
1422            ACT,,XX(II).EQ.7,STRP;        A/C 2 SYM ABORT
1423            ACT,,XX(II).EQ.2.OR.
1424                 XX(II).EQ.5,STA2;        A/C 2 PART OF TWO SHIP
1425            ACT,,XX(II).EQ.0,T3UP;        A/C 2 STILL IN THREE SHIP
1426        ;
1427         STU1 ASSIGN,II=ATRIB(46),
1428                  XX(95)=USERF(124),2;
1429            ACT,,XX(II).EQ.5.OR.XX(II).EQ.7.OR.
1430                 XX(II).EQ.9.OR.
1431                 XX(II).EQ.11,STU2;       A/C 3 FAILED
1432            ACT,,XX(II).EQ.6,STRP;        A/C 3 SYM ABORT
1433            ACT,,XX(II).EQ.0,T3UP;        A/C 3 STILL IN THREE SHIP
1434            ACT,,XX(II).EQ.2.AND.
1435                 ATRIB(43).NE.1,STU5;     NOW A/C 1 IN TWO SHIP (CASE.NE.1)
1436            ACT,,XX(II).EQ.4,STU6;        NOW A/C 2 IN TWO SHIP (CASE.NE.1)
1437            ACT,,XX(II).EQ.2.AND.
1438                 ATRIB(43).EQ.1,PSEP;     SEND PILOT 3 BACK
1439            ACT,,XX(II).EQ.2.AND.
1440                 ATRIB(43).EQ.1,STU7;     A/C 3 NOW LEAD IN TWO SHIP (CASE 1)
1441        ;
1442         STU2 ASSIGN,XX(II)=0,
1443                  XX(95)=USERF(124),2;
1444            ACT,,,PSEP;                   SEPARATE PILOT FROM A/C
1445            ACT,,,S2LO;                   A/C 3 TO MX
1446        ;
1447         STU5 ASSIGN,ATRIB(45)=1,XX(II)=0,1;   CHANGE A/C 3 TO LEAD A/C
1448            ACT,,,STA2;
1449        ;
1450         STU6 ASSIGN,ATRIB(45)=2,XX(II)=0,1;   CHANGE A/C 3 POSITION TO 2
1451            ACT,,,STA2;
1452        ;
1453        ;                                     MOVE PILOT 1 ATTRIBUTES INTO A/C
1454        ;                                     WHICH THEN BECOMES  LEAD A/C
1455         STU7 ASSIGN,ATRIB(32)=XX(81),ATRIB(33)=XX(82),
1456                  ATRIB(34)=XX(83),ATRIB(35)=XX(84),
1457                  ATRIB(36)=XX(85),ATRIB(37)=XX(86),
1458                  ATRIB(38)=XX(87),ATRIB(39)=XX(88),
```

```
1459                    ATRIB(40)=XX(89),ATRIB(41)=XX(90),
1460                    ATRIB(42)=XX(91),1;
1461                ACT,,,STU5;
1462            ;
1463             STA2 ASSIGN,ATRIB(44)=2,
1464                    ATRIB(3)=USERF(22),
1465                    ATRIB(1)=USERF(17),1;     MAKE TWO SHIP AND UNPARK
1466                ACT,,,TMA2;
1467             T3UP ASSIGN,ATRIB(3)=USERF(22),
1468                    ATRIB(1)=USERF(17),1;
1469                ACT,,,TMA3;
1470            ;
```

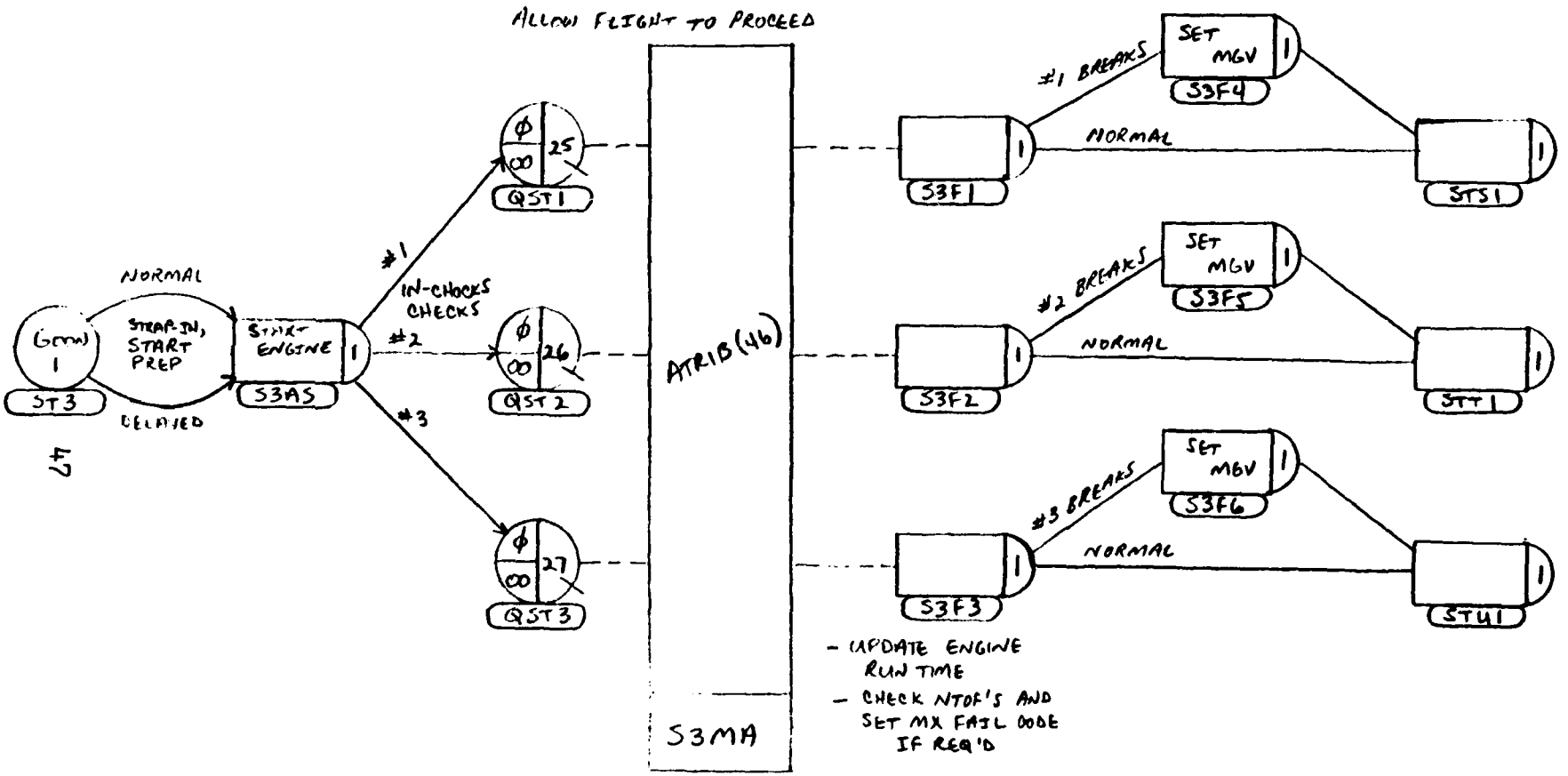


Fig A.5.1 Three-Ship Start

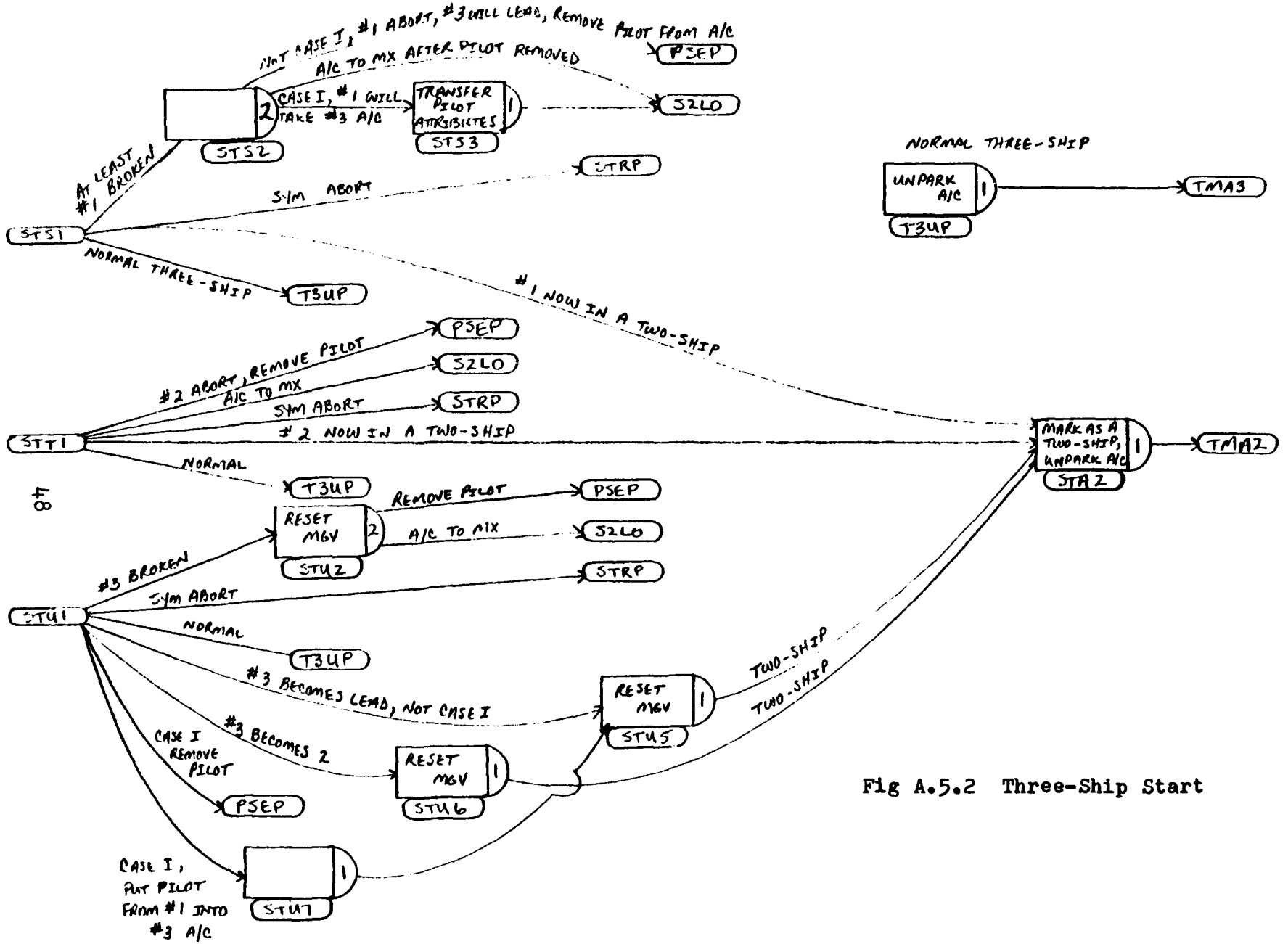


Fig A.5.2 Three-Ship Start

```
1471            ; THREE-SHIP TAXI, MARSHALL AND ARM
1472            ;    AFTER START THE A/C RELEASE THEIR MAINTENANCE TEAMS AND TAXI
1473            ; TO A MARSHALLING AREA AND THEN TO THE ARMING AREA.  A/C
1474            ; ACQUIRE AN ARMING CREW FOR ARMING SERVICE.  WHEN ARMING IS
1475            ; COMPLETED THE A/C ARE EVALUATED FOR FAILURE AND ROUTED
1476            ; ACCORDINGLY.  THE A/C PROCEED AS A THREE-SHIP, A TWO-SHIP,
1477            ; OR THE MISSION IS SCRUBBED.
1478            ;
1479             TMA3 FREE,MXTEAM/1,1;                RELEASE MX TEAM
1480                    ACT/43,USERF(63)+TRIAG(2,3,4),
1481                        DRAND.GT.XX(67),T3AS;     XX(67) - % A/C THAT DELAY TMA
1482                    ACT/44,USERF(63)+TRIAG(6,8,10),
1483                                    ,T3AS;  A/C THAT WERE DELAYED
1484            ;
1485             T3AS GOON,1;
1486                    ACT,,ATRIB(45).EQ.1,QTM1;     A/C 1
1487                    ACT,,ATRIB(45).EQ.2,QTM2;     A/C 2
1488                    ACT,,ATRIB(45).EQ.3,QTM3;     A/C 3
1489            ;
1490             QTM1 QUEUE(30),,,,T3MA;              WAIT TO REFORM FLT
1491            ;
1492             QTM2 QUEUE(31),,,,T3MA;
1493            ;
1494             QTM3 QUEUE(32),,,,T3MA;
1495            ;
1496             T3MA MATCH,46,QTM1/T3F1,QTM2/T3F2,
1497                                 QTM3/T3F3;     REFORM FLT
1498            ;                                   UPDATE MISSION STATUS CODE BY A/C
1499             T3F1 ASSIGN,ATRIB(8) =TNOW - ATRIB(8),
1500                        ATRIB(14)=ATRIB(14)+ATRIB(8),
1501                        ATRIB(7) =ATRIB(7) +ATRIB(8),
1502                        ATRIB(18)=USERF(51),
1503                        ATRIB(8) =TNOW,1;
1504                    ACT,,USERF(37).GE.2,T3F4;     A/C 1 BROKEN
1505                    ACT,.0001,,TMS1;              A/C 1 OK
1506            ;
1507             T3F4 ASSIGN,II=ATRIB(46),
1508                        XX(II)=XX(II)+2,1;
1509                    ACT,.0001,,TMS1;
1510            ;                                   UPDATE MISSION STATUS CODE BY A/C
1511             T3F2 ASSIGN,ATRIB(8) =TNOW-ATRIB(8),
1512                        ATRIB(14)=ATRIB(14)+ATRIB(8),
1513                        ATRIB(7) =ATRIB(7) +ATRIB(8),
1514                        ATRIB(18)=USERF(51),
1515                        ATRIB(8) =TNOW,1;
1516                    ACT,,USERF(37).GE.2,T3F5;     A/C 2 BROKEN
1517                    ACT,.0001,,TMT1;              A/C 2 OK
1518            ;
1519             T3F5 ASSIGN,II=ATRIB(46),
1520                        XX(II)=XX(II)+4,1;
```

```
1521                ACT,.0001,,TMT1;
1522          ;                                   UPDATE MISSION STATUS CODE BY A/C
1523           T3F3 ASSIGN,ATRIB(8) =TNOW -ATRIB(8),
1524                       ATRIB(14)=ATRIB(14)+ATRIB(8),
1525                       ATRIB(7) =ATRIB(7) +ATRIB(8),
1526                       ATRIB(18)=USERF(51),
1527                       ATRIB(8) =TNOW,1;
1528                ACT,,USERF(37).GE.2,T3F6;     A/C 3 BROKEN
1529                ACT,.0001,,TMU1;              A/C 3 OK
1530          ;
1531           T3F6 ASSIGN,II=ATRIB(46),
1532                       XX(II)=XX(II)+5,1;
1533                ACT,.0001,,TMU1;
1534          ;                                   EVALUATE MISSION CODE TO DETERMINE
1535           TMS1 ASSIGN,II=ATRIB(46),1;        STATUS OF FLIGHT (EACH A/C)
1536                ACT,,XX(II).EQ.2.OR.XX(II).EQ.6.OR.
1537                     XX(II).EQ.7.OR.
1538                     XX(II).EQ.11,T2LO;       A/C 1 FAIL
1539                ACT,,XX(II).EQ.9,TMZ2;        A/C 1 SYM ABORT
1540                ACT,,XX(II).EQ.4.OR.
1541                     XX(II).EQ.5,TMZ3;        A/C 1 NOW IN A TWO SHIP
1542                ACT,,XX(II).EQ.0,TO3;         A/C 1 STILL IN A THREE SHIP
1543
1544           T2LO ASSIGN,ATRIB(13)=7,
1545                       XX(95)=USERF(124),2;
1546                ACT,USERF(67),USERF(12).EQ.1,PSEP;
1547                ACT,,USERF(12).EQ.1,T2LR;
1548                ACT,,,MAIN;
1549          ;
1550           T2LR ASSIGN,ATRIB(32)=0,1;
1551                ACT,,,MAIN;
1552          ;
1553           TMZ2 ASSIGN,ATRIB(13)=7,1;
1554                ACT,,,DEA6;                   GO TO REFUEL
1555
1556           TMT1 ASSIGN,II=ATRIB(46),1;
1557                ACT,,XX(II).EQ.4.OR.XX(II).EQ.6.OR.
1558                     XX(II).EQ.9.OR.
1559                     XX(II).EQ.11,T2LO;       A/C 2 FAIL
1560                ACT,,XX(II).EQ.7.OR.XX(II).EQ.2.AND.
1561                     ATRIB(43).EQ.1,TMZ2;     A/C 2 SYM ABORT
1562                ACT,,XX(II).EQ.5.OR.XX(II).EQ.2.AND.
1563                     ATRIB(43).NE.1,TMZ3;     A/C 2 PART OF A TWO SHIP
1564                ACT,,XX(II).EQ.0,TO3;         A/C 2 STILL IN A THREE SHIP
1565
1566           TMU1 ASSIGN,II=ATRIB(46),1;
1567                ACT,,XX(II).EQ.5.OR.XX(II).EQ.7.OR.
1568                     XX(II).EQ.9.OR.
1569                     XX(II).EQ.11,TMU2;       A/C 3 FAIL
1570                ACT,,XX(II).EQ.6.OR.XX(II).EQ.2.AND.
```

```
1571                    ATRIB(43).EQ.1,TMZ2;     A/C 3 SYM ABORT
1572               ACT,,XX(II).EQ.2.AND.
1573                    ATRIB(43).NE.1,TMU4;     A/C 3 FLT LEAD OF A TWO SHIP
1574               ACT,,XX(II).EQ.4,TMU5;        A/C 3 PART OF A TWO SHIP
1575               ACT,,XX(II).EQ.0,TO3;         A/C 3 STILL IN A THREE SHIP
1576
1577          TMU2 ASSIGN,XX(II)=0,1;            A/C TO MX
1578               ACT,,,T2LO;
1579
1580          TMU4 ASSIGN,ATRIB(45)=1,XX(II)=0,1; CHANGE A/C 3 TO LEAD A/C
1581               ACT,,,TMZ3;
1582
1583          TMU5 ASSIGN,ATRIB(45)=2,XX(II)=0,1; CHANGE A/C 3 TO POSITION 2
1584               ACT,,,TMZ3;
1585
1586          TMZ3 ASSIGN,ATRIB(44)=2,1;         CHANGE FLT TO A TWO-SHIP
1587               ACT,,,TO2;
```

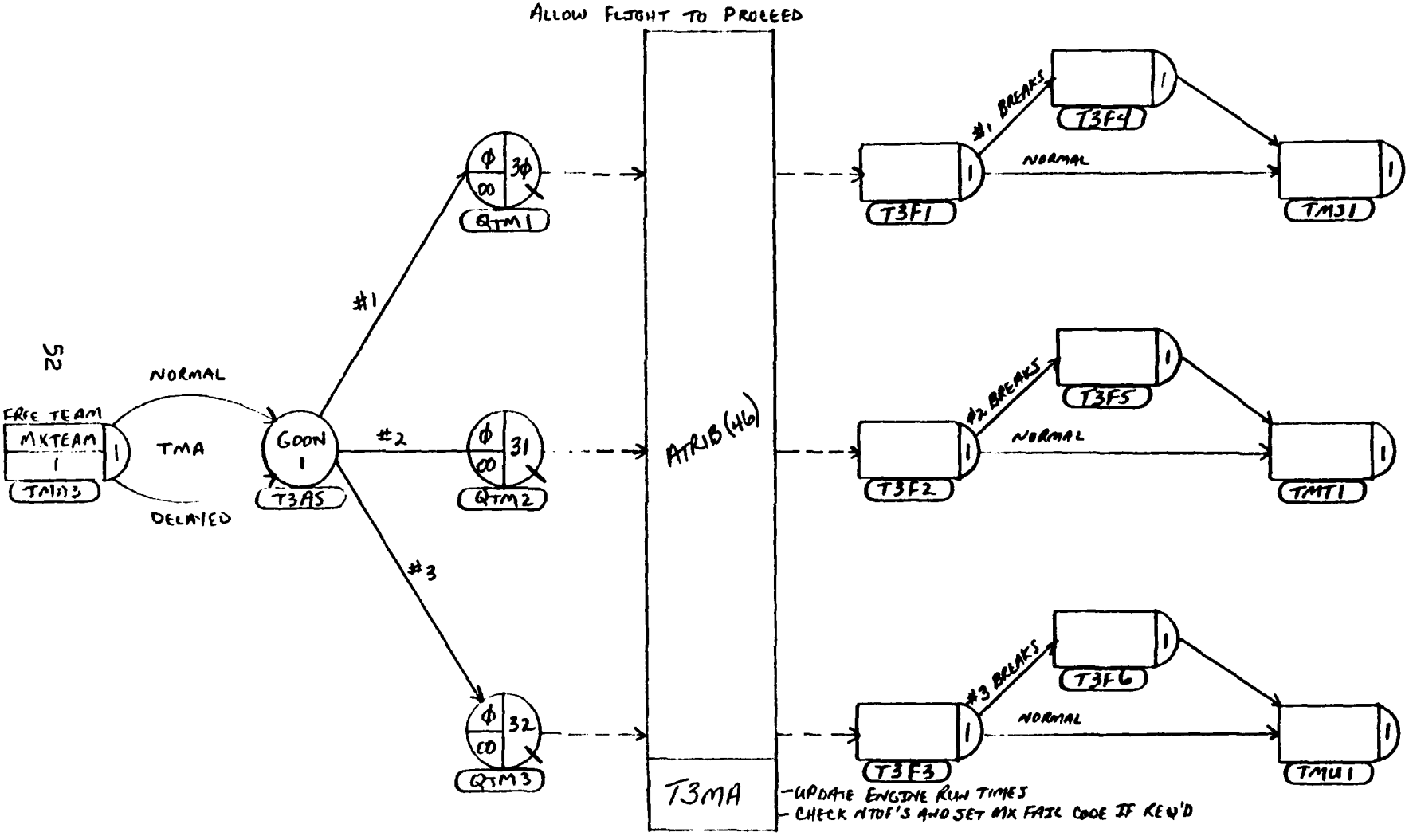


Fig A.6.1 Three-Ship Taxi, Marshall, and Arm

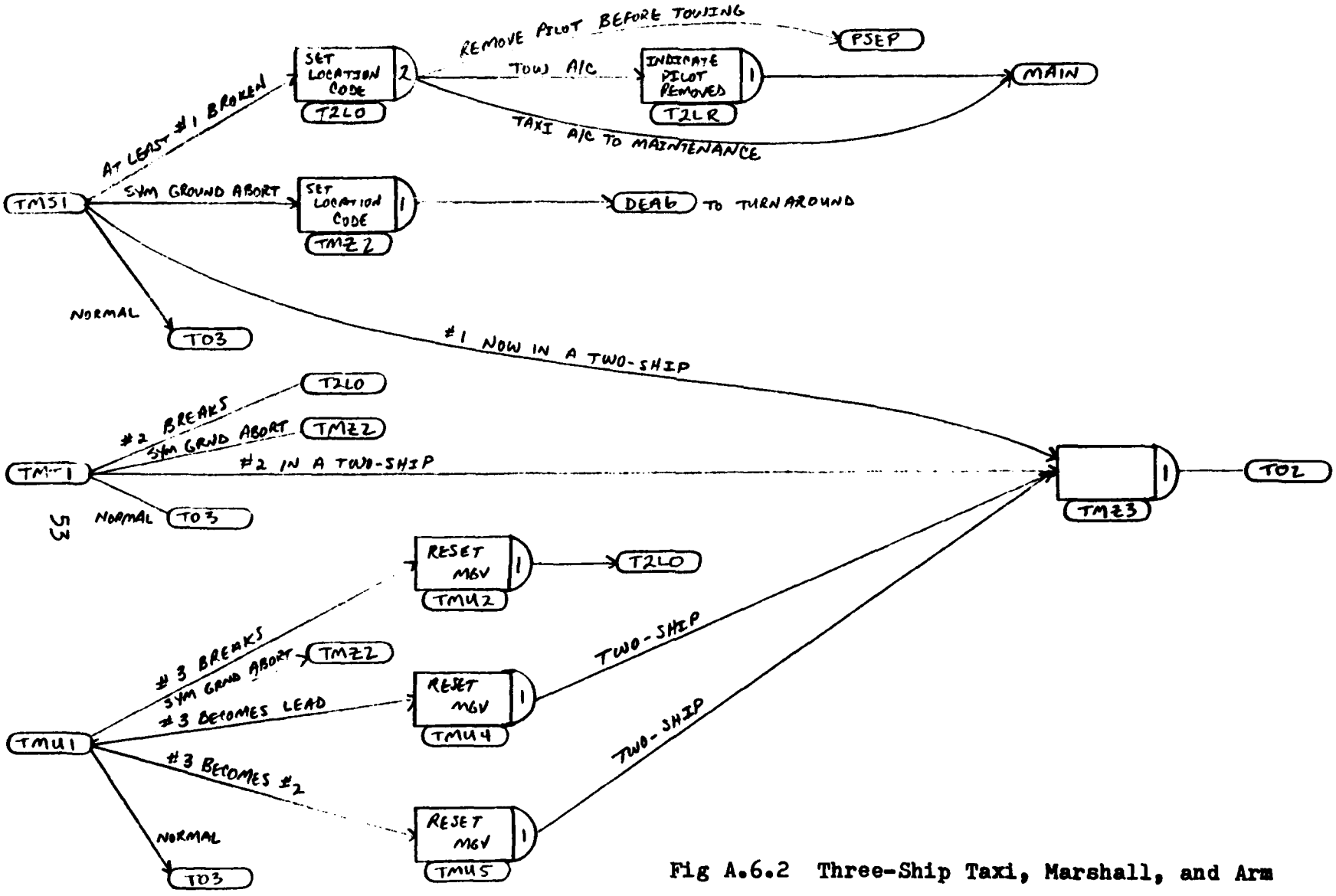


Fig A.6.2 Three-Ship Taxi, Marshall, and Arm

```
1588            ; THREE-SHIP TAKEOFF
1589           ;    THE LEAD A/C ACQUIRES THE RUNWAY FOR THE FLIGHT AND THE FLIGHT
1590           ; TAXIS ON THE RUNWAY FOR RUN-UP CHECKS AND TAKEOFF.  THERE IS A
1591           ; POSSIBILITY OF DELAY. A/C ARE EVALUATED FOR FAILURES AND THERE
1592           ; IS THE ENTIRE GAMUT OF POSSIBILITIES FROM SIMPLE MAINTENANCE
1593           ; ABORT OF ONE A/C TO SYMPATHETIC AIR ABORT.  IF ALL 3 A/C LAUNCH
1594           ; THEY PROCEED TO REJOIN (REJ3), IF ONLY 2 THEY REJOIN (REJ2),
1595           ; ELSE THE MISSION IS SCRUBBED.
1596           ;
1597            TO3  GOON,1;                          ROUTINE FOR FLT TO GET THE RNWY
1598                   ACT,,ATRIB(45).EQ.1,TOA3;
1599                   ACT,,,TOC3;
1600           ;
1601            TOA3 AWAIT(45),RUNWAY/1,1;
1602                   ACT,.0001,DRAND.GT.XX(68),TOC3;
1603                   ACT,TRIAC(.5,1,2),,TOC3;
1604           ;
1605            TOC3 GOON,1;
1606                   ACT,,ATRIB(45).EQ.1,QTO1;
1607                   ACT,,ATRIB(45).EQ.2,QTO2;
1608                   ACT,,ATRIB(45).EQ.3,QTO3;
1609           ;
1610           ;
1611            QTO1 QUEUE(35),,,,L3MA;               WAIT TO REFORM FLIGHT
1612           ;
1613            QTO2 QUEUE(36),,,,L3MA;
1614           ;
1615            QTO3 QUEUE(37),,,,L3MA;
1616           ;
1617            L3MA MATCH,46,QTO1/L3F1,
1618                   QTO2/L3F2,QTO3/L3F3;           REFORM FLIGHT
1619           ;                                     UPDATE MISSION STATUS CODE BY A/C
1620            L3F1 ASSIGN,ATRIB(8)=TNOW-ATRIB(8),
1621                   ATRIB(8)=ATRIB(8)+1,
1622                   ATRIB(14)=ATRIB(14)+ATRIB(8),
1623                   ATRIB(7)=ATRIB(7)+ATRIB(8),
1624                   ATRIB(18)=USERF(51),
1625                   ATRIB(8)=TNOW,1;
1626                   ACT,,USERF(37).GE.2,L3F4;     A/C 1 BROKEN
1627                   ACT,.0001,,TOS1;              A/C 1 OK
1628           ;
1629            L3F4 ASSIGN,II=ATRIB(46),
1630                   XX(II)=XX(II)+2,1;
1631                   ACT,.0001,,TOS1;
1632           ;                                     UPDATE MISSION STATUS CODE BY A/C
1633            L3F2 ASSIGN,ATRIB(8)=TNOW-ATRIB(8),
1634                   ATRIB(8)=ATRIB(8)+1,
1635                   ATRIB(14)=ATRIB(14)+ATRIB(8),
1636                   ATRIB(7)=ATRIB(7)+ATRIB(8),
1637                   ATRIB(18)=USERF(51),
```

```
                ATRIB(8)=TNOW,1;
          ACT,,USERF(37).GE.2,L3F5;     A/C 2 BROKEN
          ACT,.0001,,TOT1;              A/C 2 OK
;
 L3F5 ASSIGN,II=ATRIB(46),
          XX(II)=XX(II)+4,1;
          ACT,.0001,,TOT1;
;                                       UPDATE MISSION STATUS CODE BY A/C
 L3F3 ASSIGN,ATRIB(8)=TNOW-ATRIB(8),
          ATRIB(8)=ATRIB(8)+1,
          ATRIB(14)=ATRIB(14)+ATRIB(8),
          ATRIB(7)=ATRIB(7)+ATRIB(8),
          ATRIB(18)=USERF(51),
          ATRIB(8)=TNOW,1;
          ACT,,USERF(37).GE.2,L3F6;     A/C 3 BROKEN
          ACT,.0001,,TOU1;              A/C 3 OK
;
 L3F6 ASSIGN,II=ATRIB(46),
          XX(II)=XX(II)+5,1;
          ACT,.0001,,TOU1;
;
 TOS1 ASSIGN,II=ATRIB(46),1;           EVALUATE A/C 1 STATUS
          ACT,1,XX(II).EQ.2.OR.XX(II).EQ.6.OR.
               XX(II).EQ.7.OR.
               XX(II).EQ.11,M3CA;       A/C 1 BROKEN ON RUNWAY
          ACT,1,XX(II).EQ.9,FRS3;       A/C 1 SYM AIR ABORT
          ACT,1,XX(II).EQ.4.OR.
              XX(II).EQ.5,FRT2;         A/C 1 PART OF A TWO SHIP
          ACT,,XX(II).EQ.0,FRW3;        A/C 1 STILL IN A THREE SHIP
;
 TOT1 ASSIGN,II=ATRIB(46),1;           EVALUATE A/C 2 STATUS
          ACT,1,XX(II).EQ.4.OR.XX(II).EQ.6.OR.
          XX(II).EQ.9.OR.
          XX(II).EQ.11,DEA3;            A/C 2 BROKEN ON RUNWAY
          ACT,,XX(II).EQ.7.AND.
              ATRIB(43).NE.1,TOFY;      A/C 2 SYM AIR ABORT
          ACT,1,XX(II).EQ.2.AND.ATRIB(43).EQ.1.OR.
               XX(II).EQ.7.AND.
               ATRIB(43).EQ.1,TOZ2;     A/C 2 SYM GRND ABORT
          ACT,,XX(II).EQ.5.OR.XX(II).EQ.2.AND.
               ATRIB(43).NE.1,TOZ1;     A/C 2 PART OF A TWO SHIP
          ACT,,XX(II).EQ.0,REJ3;        A/C 2 STILL IN A THREE SHIP
;
 TOU1 ASSIGN,II=ATRIB(46),1;           EVALUATE STATUS OF A/C 3
          ACT,,XX(II).EQ.5.OR.XX(II).EQ.7.OR.
               XX(II).EQ.9.OR.
               XX(II).EQ.11,TOU2;       A/C 3 BROKEN ON RUNWAY
          ACT,1,XX(II).EQ.6.OR.XX(II).EQ.2.AND.
               ATRIB(43).EQ.1,TOZ2;     A/C 3 SYM GRND ABORT
          ACT,,XX(II).EQ.2.AND.
```

```
                        ATRIB(43).NE.1,TOU4;    A/C 3 IS NOW FLT LEAD IN A TWO SHIP
              ACT,,XX(II).EQ.4,TOU5;            A/C 3 NOW IN A TWO SHIP
              ACT,,XX(II).EQ.0,REJ3;            A/C 3 STILL IN A THREE SHIP
;
 TOU2 ASSIGN,XX(II)=0,1;
              ACT,1,,DEA3;
;
 TOU4 ASSIGN,ATRIB(45)=1,XX(II)=0,1;    CHANGE A/C 3 TO FLT LEAD (A/C 1)
              ACT,,,TOZ1;
;
 TOU5 ASSIGN,ATRIB(45)=2,XX(II)=0,1;    CHANGE A/C 3 TO A/C POSITION 2
              ACT,,,TOZ1;
;
 TOZ1 ASSIGN,ATRIB(44)=2,1;             CHANGE NUMBER OF A/C IN MSN TO 2
              ACT,,,REJ2;
;
 FRW3 FREE,RUNWAY/1,1;                  RELEASE RUNWAY - REJOIN THREE SHIP
              ACT,,,REJ3;
;
 FRT2 FREE,RUNWAY/1,1;                  RELEASE RUNWAY - REJOIN TWO SHIP
              ACT,,,TOZ1;
;
 M3GA FREE,RUNWAY/1,1;                  RELEASE RUNWAY - GO TO DEARM
              ACT,,,DEA3;
;
 TOZ2 ASSIGN,ATRIB(17)=1,1;             SYM GRND ABORT
              ACT,,,DEA3;
;
 FRS3 FREE,RUNWAY/1,1;                  SYM AIR ABORT
              ACT,,,TOFY;
;
```

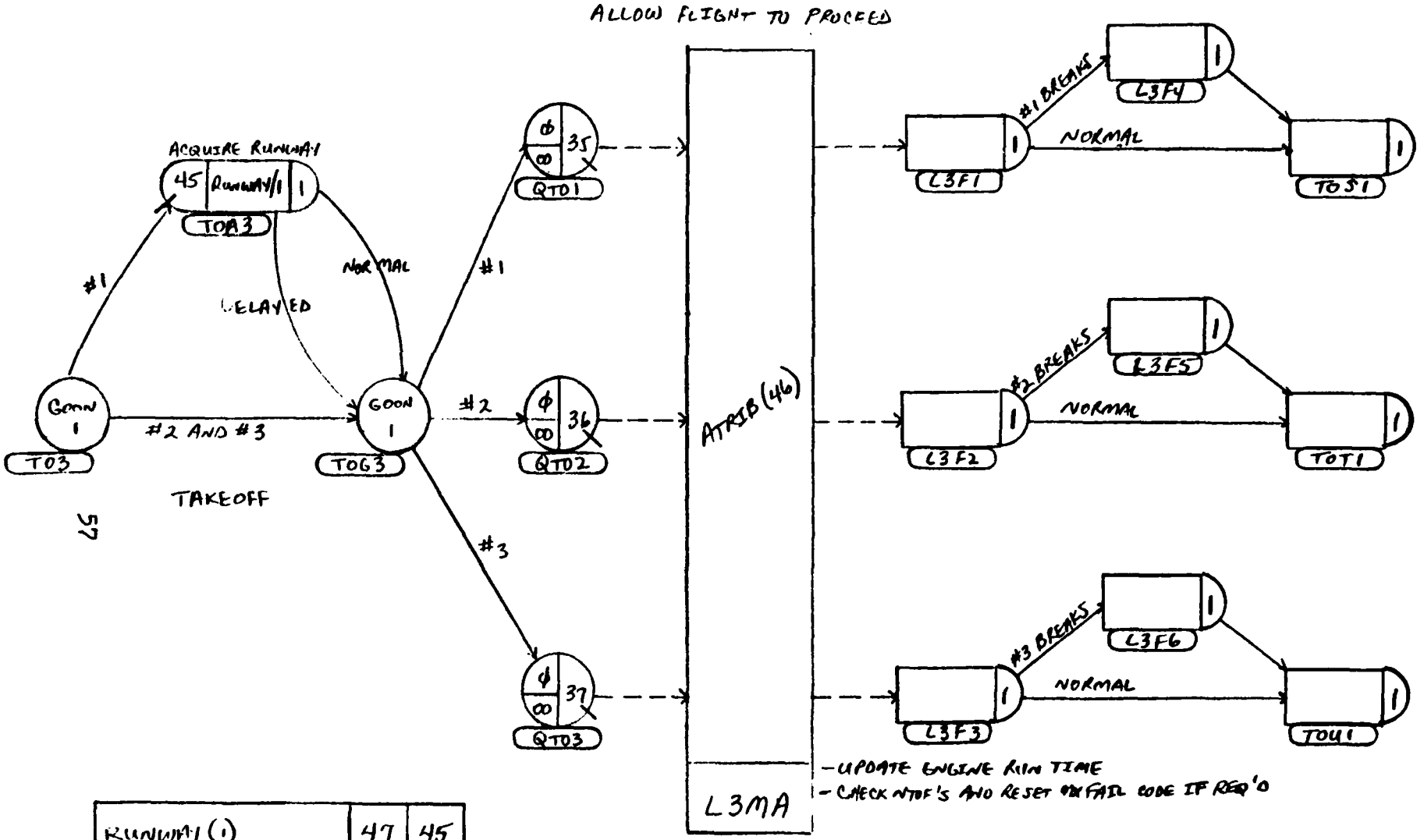


Fig A.7.1 Three-Ship Takeoff

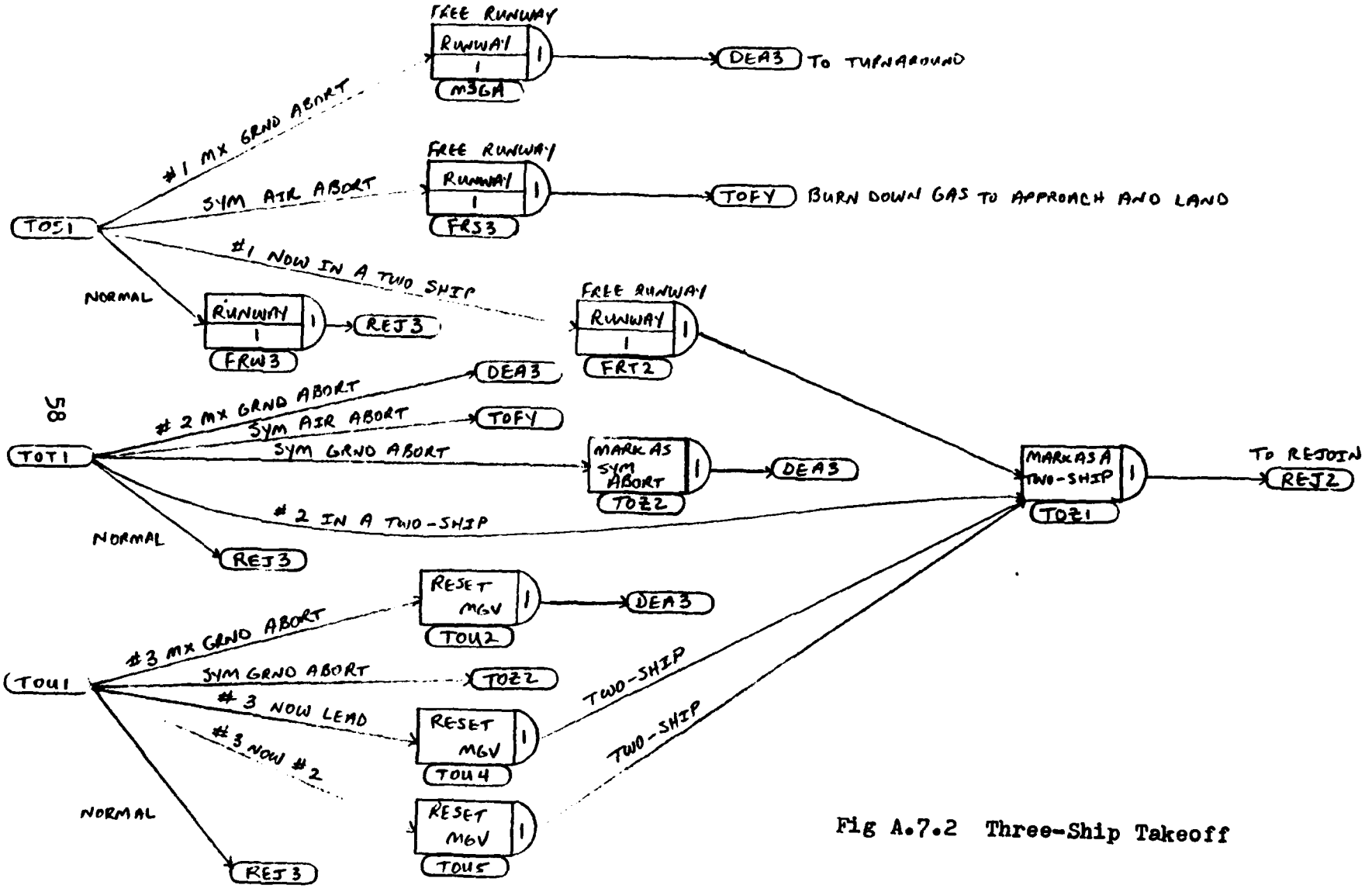


Fig A.7.2 Three-Ship Takeoff

```
1719            ; THREE-SHIP REJOIN
1720          ;    AFTER TAKEOFF THE A/C REJOIN BEFORE PROCEEDING ON THEIR MISSION.
1721          ; A/C ARE EVALUATED FOR FAILURES BEFORE BEING ALLOWED TO PROCEED
1722          ; ON THE MISSION.  BROKEN A/C ARE ROUTED ACCORDINGLY.  THE RE-
1723          ; SULT MAY BE A THREE-SHIP MISSION, A TWO-SHIP MISSION, OR THE
1724          ; THE MISSION MAY BE SCRUBBED WITH A SYMPATHETIC AIR ABORT.  IF
1725          ; AN AIRCRAFT AIR ABORTS IT MAY JETTISON AND/OR BURN DOWN GAS
1726          ; PRIOR TO LANDING.  AIRCRAFT WHICH EXPERIENCE FAILURES IN THE
1727          ; AIR WILL CRASH IF THE FAILURES ARE SEVERE ENOUGH IN CERTAIN
1728          ; SYSTEMS OR COMBINATIONS OF SYSTEMS.  SEE THE FORTRAN LISTING
1729          ; FOR THE SPECIFIC FAILURE LEVELS AND COMBINATIONS OF SYSTEMS.
1730          ;
1731           REJ3 ASSIGN,XX(95)=USERF(121),
1732                       XX(94)=XX(94) + 1,
1733                       XX(95)=USERF(122),1;
1734                 ACT,TRIAG(1,2,3),
1735                     DRAND.GT.XX(69),R3ST;     NORMAL REJOIN
1736                 ACT,TRIAG(2,3,4),,R3ST;       DELAY IN REJOIN
1737          ;
1738           R3ST GOON,1;
1739                 ACT,,ATRIB(45).EQ.1,QRE1;
1740                 ACT,,ATRIB(45).EQ.2,QRE2;
1741                 ACT,,ATRIB(45).EQ.3,QRE3;
1742          ;
1743           QRE1 QUEUE(40),,,,R3MA;             WAIT TO REFORM FLIGHT
1744          ;
1745           QRE2 QUEUE(41),,,,R3MA;
1746          ;
1747           QRE3 QUEUE(42),,,,R3MA;
1748          ;                                   UPDATE MISSION STATUS CODE BY A/C
1749           R3MA MATCH,46,QRE1/R3F1,
1750                 QRE2/R3F2,QRE3/R3F3;
1751          ;
1752           R3F1 ASSIGN,ATRIB(8)=TNOW-ATRIB(8),
1753                 ATRIB(15)=ATRIB(8),
1754                 ATRIB(7)=ATRIB(7)+ATRIB(8),
1755                 ATRIB(18)=USERF(51),
1756                 ATRIB(8)=TNOW,1;
1757                 ACT,,USERF(37).GE.2,R3F4;     A/C 1 BROKEN
1758                 ACT,.0001,,RES1;              A/C 1 OK
1759          ;                                   UPDATE MISSION STATUS CODE BY A/C
1760           R3F4 ASSIGN,II=ATRIB(46),
1761                 XX(II)=XX(II)+2,1;            SET UP TO GO TO MX
1762                 ACT,.0001,,RES1;
1763          ;
1764           R3F2 ASSIGN,ATRIB(8)=TNOW-ATRIB(8),
1765                 ATRIB(15)=ATRIB(8),
1766                 ATRIB(7)=ATRIB(7)+ATRIB(8),
1767                 ATRIB(18)=USERF(51),
1768                 ATRIB(8)=TNOW,1;
```

```
1769                    ACT,,USERF(37).GE.2,R3F5;     A/C 2 BROKEN
1770                  ACT,.0001,,RET1;              A/C 2 OK
1771            ;                                   UPDATE MISSION STATUS CODE BY A/C
1772             R3F5 ASSIGN,II=ATRIB(46),
1773                   XX(II)=XX(II)+4,1;           SET UP TO GO TO MX
1774                  ACT,.0001,,RET1;
1775            ;
1776             R3F3 ASSIGN,ATRIB(8)=TNOW-ATRIB(8),
1777                  ATRIB(15)=ATRIB(8),
1778                  ATRIB(7)=ATRIB(7)+ATRIB(8),
1779                  ATRIB(18)=USERF(51),
1780                  ATRIB(8)=TNOW,1;
1781                  ACT,,USERF(37).GE.2,R3F6;     A/C 3 BROKEN
1782                  ACT,.0001,,REU1;              A/C 3 OK
1783            ;
1784             R3F6 ASSIGN,II=ATRIB(46),
1785                  XX(II)=XX(II)+5,1;            SET UP TO GO TO MX
1786                  ACT,.0001,,REU1;
1787            ;
1788             RES1 ASSIGN,II=ATRIB(46),1;
1789                  ACT,,XX(II).EQ.2.OR.XX(II).EQ.6.OR.
1790                       XX(II).EQ.7.OR.
1791                       XX(II).EQ.11,REFY;       GO TO MX
1792                  ACT,,XX(II).EQ.9,R2S3;        SYM AIR ABORT
1793                  ACT,,XX(II).EQ.4.OR.
1794                       XX(II).EQ.5,REZ1;        REJOIN AS TWO-SHIP
1795                  ACT,.0001,XX(II).EQ.0,MISS;   THREE-SHIP MISSION
1796            ;
1797             RET1 ASSIGN,II=ATRIB(46),1;
1798                  ACT,,XX(II).EQ.4.OR.XX(II).EQ.6.OR.
1799                       XX(II).EQ.9.OR.
1800                       XX(II).EQ.11,REFY;       GO TO MX
1801                  ACT,,XX(II).EQ.7.OR.XX(II).EQ.2.AND.
1802                       ATRIB(43).EQ.1,R2S3;     SYM AIR ABORT
1803                  ACT,,XX(II).EQ.5.OR.XX(II).EQ.2.AND.
1804                       ATRIB(43).NE.1,REZ1;     REJOIN AS TWO-SHIP
1805                  ACT,.0001,XX(II).EQ.0,MISS;   THREE-SHIP MISSION
1806            ;
1807             REU1 ASSIGN,II=ATRIB(46),1;
1808                  ACT,,XX(II).EQ.5.OR.XX(II).EQ.7.OR.
1809                       XX(II).EQ.9.OR.
1810                       XX(II).EQ.11,REU2;       GO TO MX
1811                  ACT,,XX(II).EQ.6.OR.XX(II).EQ.2.AND.
1812                       ATRIB(43).EQ.1,R2S3;     SYM AIR ABORT
1813                  ACT,,XX(II).EQ.2.AND.
1814                       ATRIB(43).NE.1,REU4;     BECOME LEAD OF TWO-SHIP
1815                  ACT,,XX(II).EQ.4,REU5;        BECOME 2 IN A TWO-SHIP
1816                  ACT,.0001,XX(II).EQ.0,MISS;   THREE-SHIP MISSION
1817            ;
1818             REU2 ASSIGN,XX(II)=0,1;
```

```
1819                    ACT,,,REFY;                 GO TO MX
1820            ;
1821             REU4 ASSIGN,ATRIB(45)=1,XX(II)=0,1;   CHANGE A/C 3 TO LEAD OF TWO-SHIP
1822                    ACT,,,REZ1;
1823            ;
1824             REU5 ASSIGN,ATRIB(45)=2,XX(II)=0,1;   CHANGE A/C 3 TO 2 IN A TWO-SHIP
1825                    ACT,,,REZ1;
1826             REZ1 ASSIGN,ATRIB(44)=2,1;            CHANGE FLT TO A TWO-SHIP
1827                    ACT,.0001,,MISS;
1828            ;
```

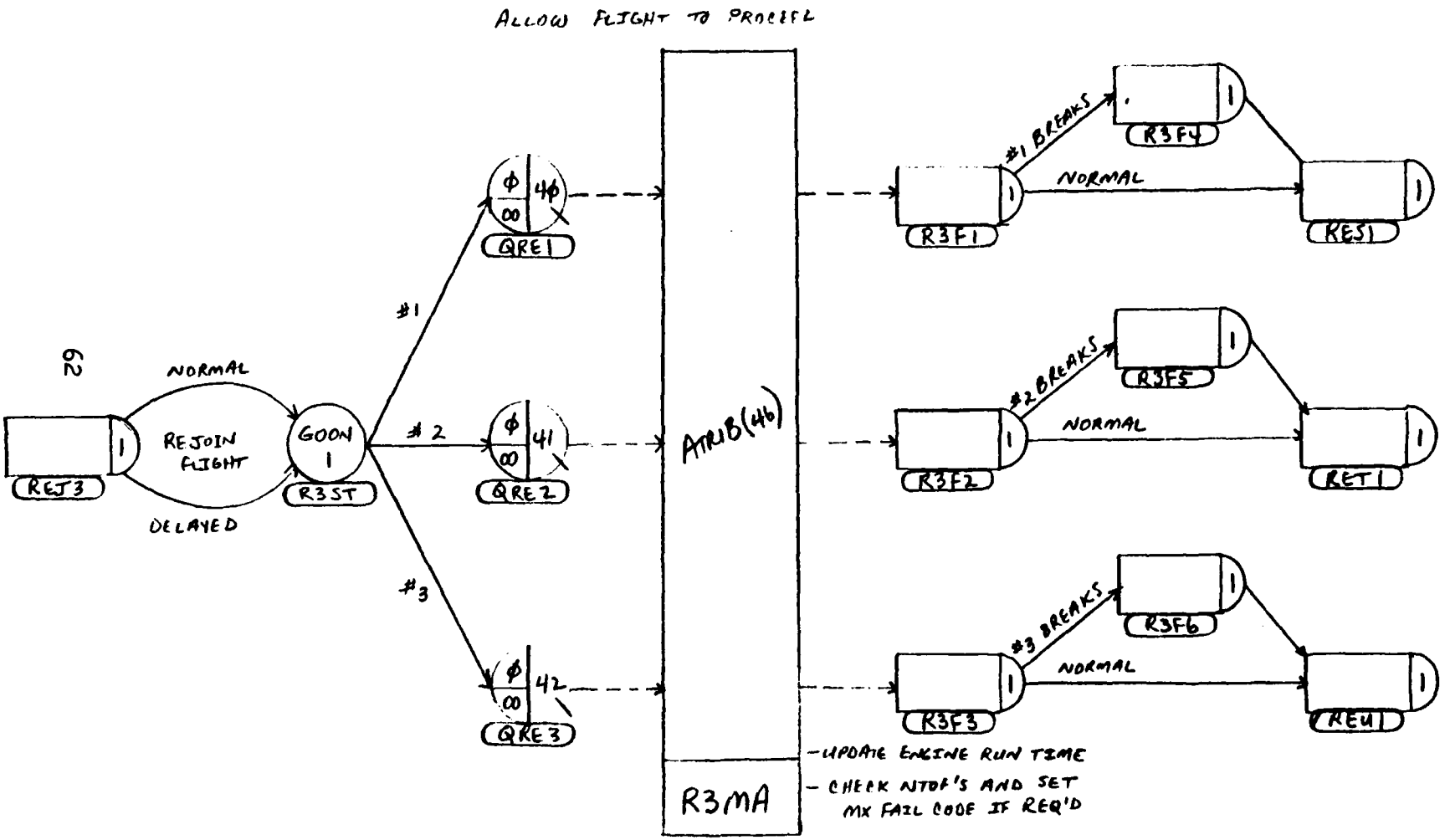


Fig A.8.1 Three-Ship Rejoin

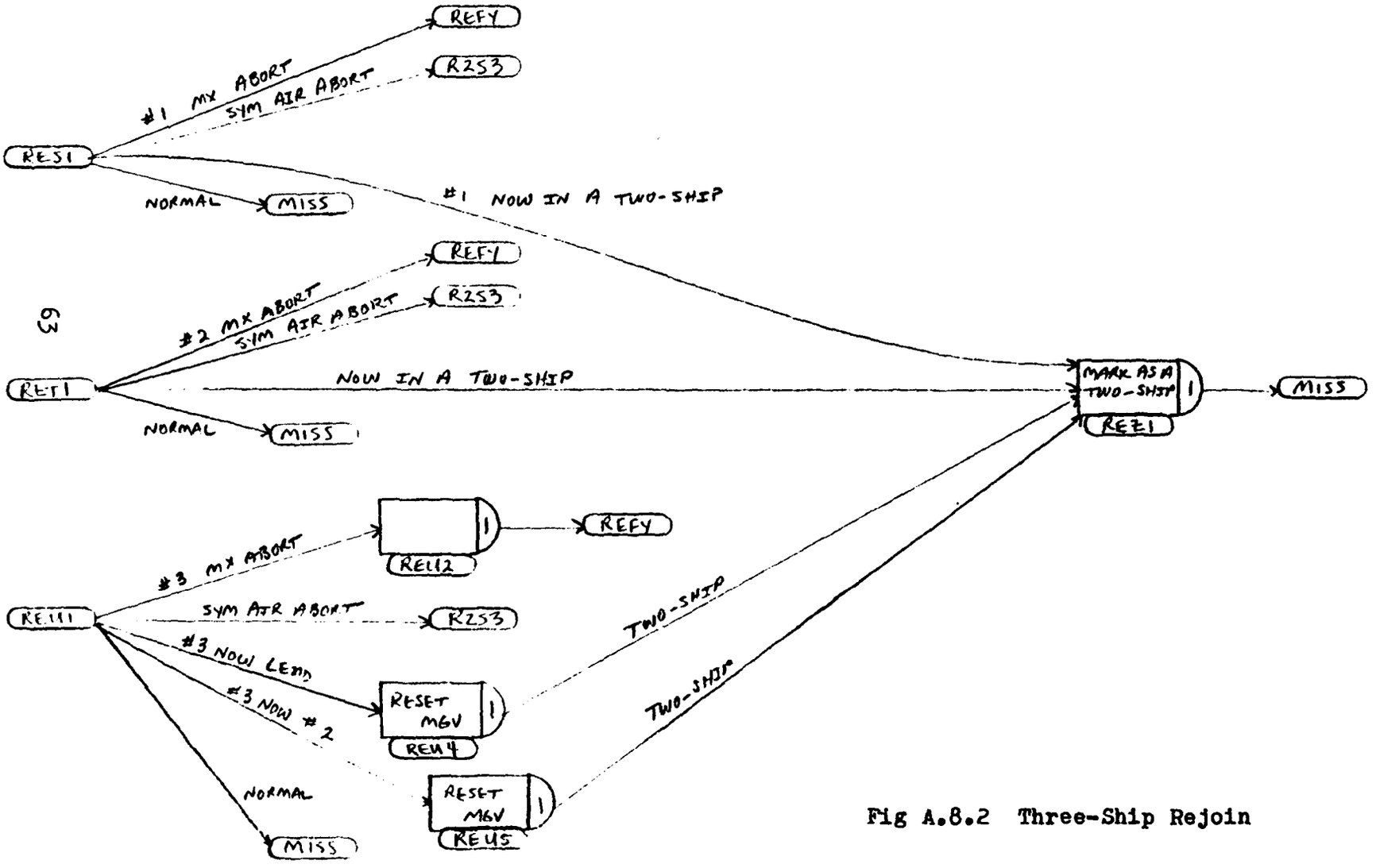


Fig A.8.2 Three-Ship Rejoin

```
1829          ; TWO-SHIP START
1830          ;    TIME IS PROVIDED FOR STRAP-IN AND PRE-START COCKPIT CHECKS. A/C
1831          ; THEN ARE STARTED AND THE IN THE CHOCKS CHECKS ARE PERFORMED.
1832          ; ONCE ENGINES ARE STARTED THE ENGINE RUN TIME IS TRACKED SO THE
1833          ; AMOUNT OF RUN TIME CAN BE COMPARED TO NTOF TO DETERMINE WHEN
1834          ; AN A/C FAILS. AFTER ALL CHECKS THE A/C ARE EVALUATED FOR ANY
1835          ; FAILURES AND ROUTED ACCORDINGLY.  IF ONE ABORTS THE OTHER IS A
1836          ; SYMPATHETIC ABORT.
1837          ;
1838           ST2  GOON,1;
1839                  ACT,TRIAG(2,3,4),
1840                   DRAND.GT.XX(66),S2AS;        NORMAL ACTIVITY DURATION
1841                  ACT,TRIAG(4,5,6),,S2AS;       DELAYED ACTIVITY DURATION
1842          ;
1843           S2AS ASSIGN,ATRIB(8)=TNOW,1;         START OF A/C OPERATIONS TIME
1844                  ACT,TRIAG(2,3,4),
1845                  ATRIB(45).EQ.1,QST4;          IN CHOCKS PRE-TAXI CHECKS
1846                  ACT,TRIAG(2,3,4),
1847                   ATRIB(45).EQ.2,QST5;
1848          ;
1849           QST4 QUEUE(28),,,,S2MA;              WAIT TO REFORM FLT
1850          ;
1851           QST5 QUEUE(29),,,,S2MA;
1852          ;
1853           S2MA MATCH,46,QST4/S2F1,QST5/S2F2;   REFORM FLT
1854          ;
1855           S2F1 ASSIGN,ATRIB(8)=TNOW-ATRIB(8),
1856                  ATRIB(14)=ATRIB(14)+ATRIB(8),
1857                  ATRIB(7)=ATRIB(7)+ATRIB(8),
1858                  ATRIB(18)=USERF(51),
1859                  ATRIB(8)=TNOW,1;              UPDATE A/C OPERATING TIME
1860                  ACT,,USERF(37).GE.2,S2F3;     A/C 1 BROKEN
1861                  ACT,.0001,,S2S1;              A/C 1 OK
1862          ;
1863           S2F3 ASSIGN,II=ATRIB(46),
1864                  XX(II)=XX(II)+2,1;
1865                  ACT,.0001,,S2S1;
1866          ;
1867           S2F2 ASSIGN,ATRIB(8)=TNOW-ATRIB(8),
1868                  ATRIB(14)=ATRIB(14)+ATRIB(8),
1869                  ATRIB(7)=ATRIB(7)+ATRIB(8),
1870                  ATRIB(18)=USERF(51),
1871                  ATRIB(8)=TNOW,1;
1872                  ACT,,USERF(37).GE.2,S2F4;     A/C 2 BROKEN
1873                  ACT,.0001,,S2S2;              A/C 2 OK
1874          ;
1875           S2F4 ASSIGN,II=ATRIB(46),
1876                  XX(II)=XX(II)+4,1;
1877                  ACT,.0001,,S2S2;
1878          ;
```

```
 1879            S2S1 ASSIGN,II=ATRIB(46),
1880                      XX(95)=USERF(124),2;
1881                 ACT,,XX(II).EQ.0,T2UP;
1882                 ACT,,XX(II).EQ.4,STRP;
1883                 ACT,,XX(II).EQ.2.OR.
1884                      XX(II).EQ.6,S2LO;
1885                 ACT,,XX(II).EQ.2.OR.
1886                      XX(II).EQ.6,PSEP;
1887            ;
1888             S2S2 ASSIGN,II=ATRIB(46),
1889                      XX(95)=USERF(124),2;
1890                 ACT,,XX(II).EQ.4.OR.
1891                      XX(II).EQ.6,PSEP;
1892                 ACT,,XX(II).EQ.4.OR.
1893                      XX(II).EQ.6,S2LO;
1894                 ACT,,XX(II).EQ.2,STRP;
1895                 ACT,,XX(II).EQ.0,T2UP;
1896            ;
1897             S2LO ASSIGN,ATRIB(13)=ATRIB(1),
1898                      ATRIB(32)=0,1;
1899                 ACT,,,SMXC;
1900            ;
1901             T2UP ASSIGN,ATRIB(3)=USERF(22),
1902                      ATRIB(1)=USERF(17),1;
1903                 ACT,,,TMA2;
1904            ;
1905             STRP ASSIGN,XX(95)=USERF(124),2;
1906                 ACT,,,PSOC;                    PATH TO SQ READY POOL (ARP%)
1907                 ACT,,,PSEP;                    PILOT SEPARATED -
1908            ;
```

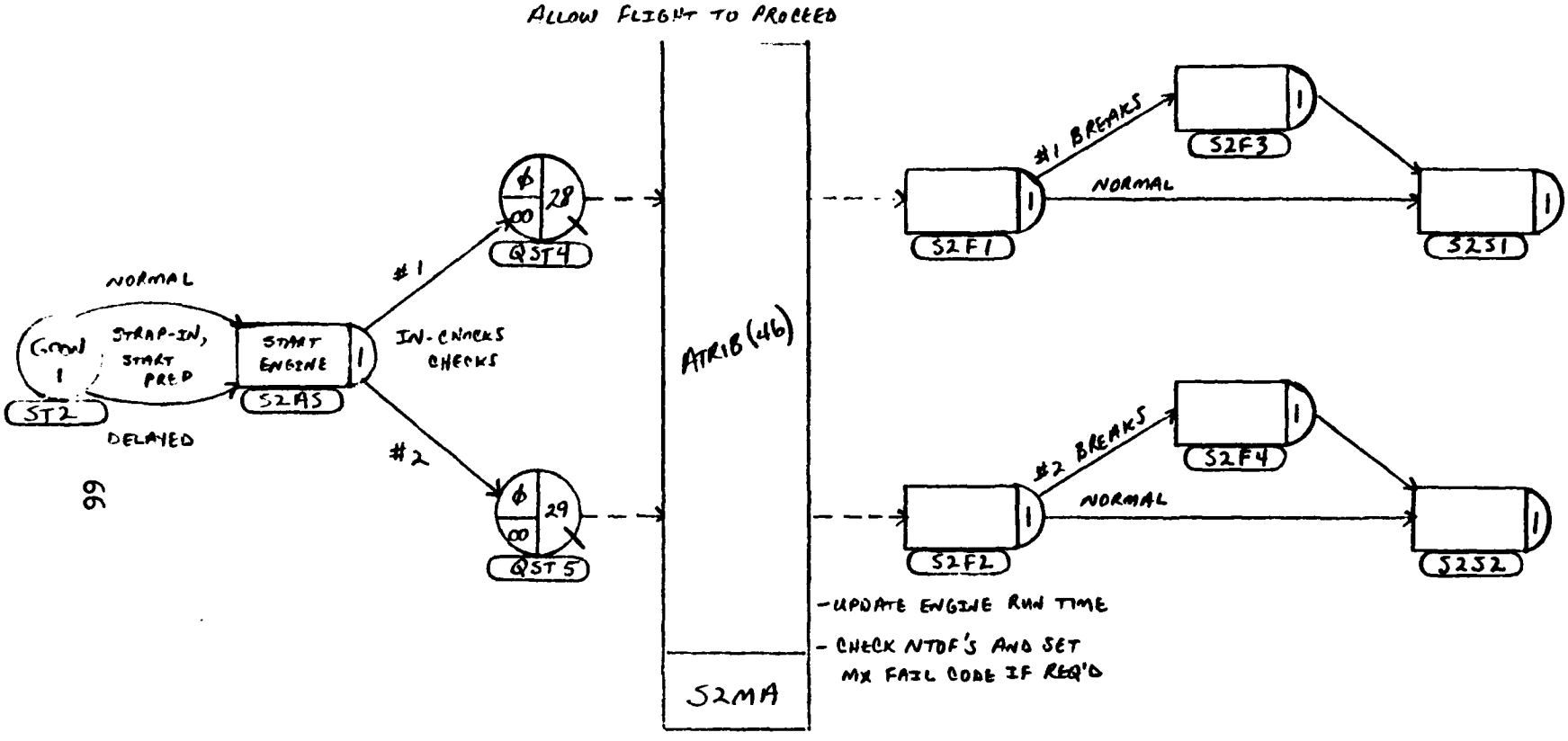


Fig A.9.1 Two-Ship Start

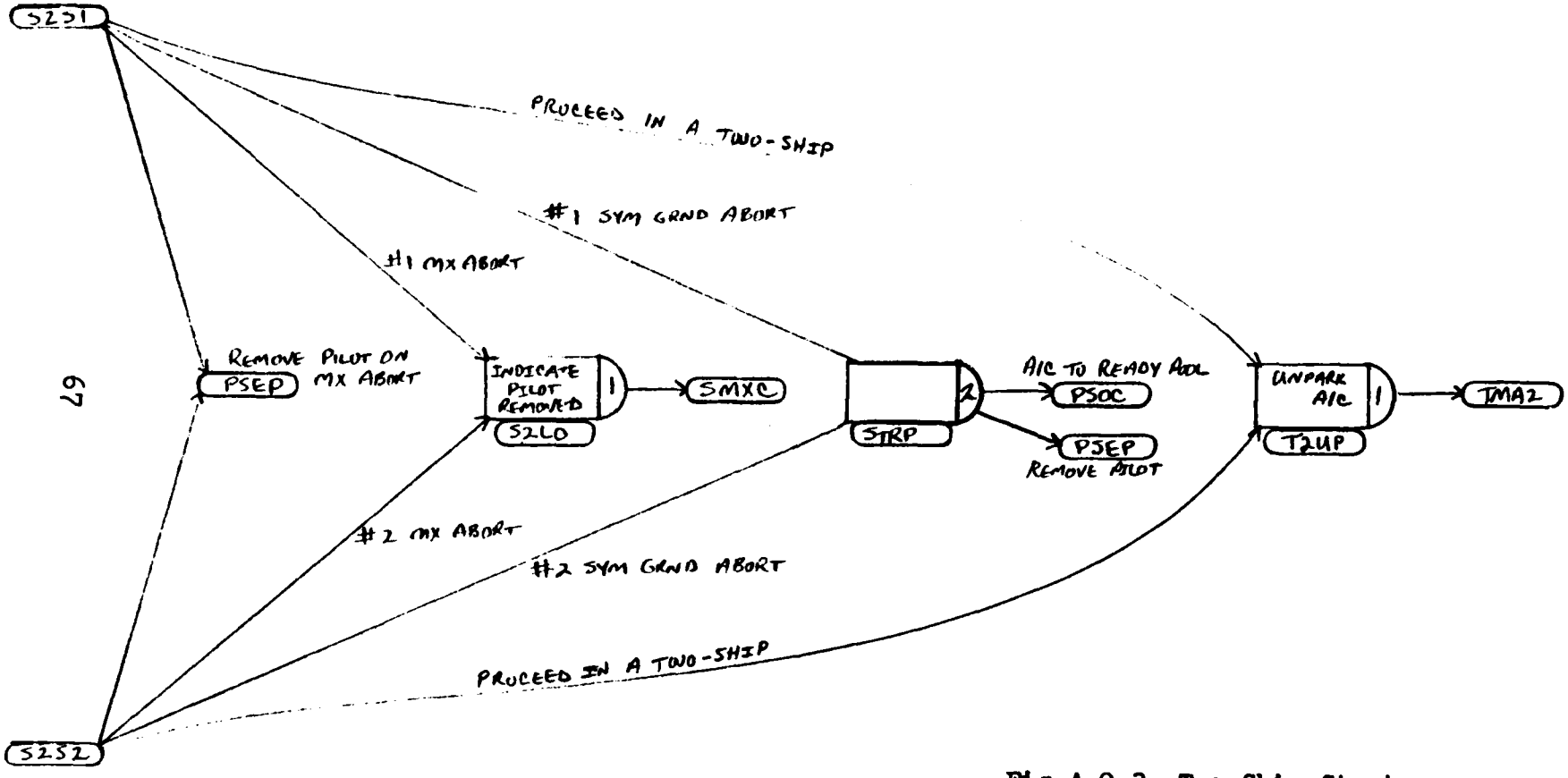


Fig A.9.2 Two-Ship Start

```
1909            ; TWO-SHIP TAXI, MARSHALL AND ARM
1910            ;    A/C FREE UP THEIR CREW CHIEF WHEN THEY TAXI TO MARSHALLING
1911            ; AND THEN ON TO THE ARMING AREA.  THE A/C ACQUIRE AN ARMING CREW
1912            ; AND ARE ARMED.  AFTER ARMING AIRCRAFT ARE EVALUATED FOR FAIL-
1913            ; URES AND ARE ROUTED ACCORDINGLY.  IF ONE A/C ABORTS THE OTHER IS A
1914            ; SYMPATHETIC ABORT.
1915            ;
1916             TMA2 FREE,MXTEAM/1,1;
1917                    ACT/45,USERF(63)+TRIAG(2,3,4),
1918                        DRAND.GT.XX(67),T2ST;     NORMAL TMA ACTIVITY DURATION
1919                    ACT/46,USERF(63)+TRIAG(6,8,10),,
1920                        T2ST;                     DELAYED TMA ACTIVITY DURATION
1921            ;
1922             T2ST GOON,1;
1923                    ACT,,ATRIB(45).EQ.1,QTM4;
1924                    ACT,,ATRIB(45).EQ.2,QTM5;
1925            ;
1926             QTM4 QUEUE(33),,,,T2MA;              WAIT TO REFORM FLIGHT
1927            ;
1928             QTM5 QUEUE(34),,,,T2MA;
1929            ;
1930             T2MA MATCH,46,QTM4/T2F1,QTM5/T2F2;   REFORM FLIGHT
1931            ;
1932             T2F1 ASSIGN,ATRIB(8)=TNOW-ATRIB(8),
1933                     ATRIB(14)=ATRIB(14)+ATRIB(8),
1934                     ATRIB(7)=ATRIB(7)+ATRIB(8),
1935                     ATRIB(18)=USERF(51),
1936                     ATRIB(8)=TNOW,1;
1937                    ACT,,USERF(37).GE.2,T2F3;     A/C 1 BROKEN
1938                    ACT,.0001,,T2S1;              A/C 1 OK
1939            ;
1940             T2F3 ASSIGN,II=ATRIB(46),
1941                     XX(II)=XX(II)+2,1;
1942                    ACT,.0001,,T2S1;
1943            ;
1944             T2F2 ASSIGN,ATRIB(8)=TNOW-ATRIB(8),
1945                     ATRIB(14)=ATRIB(14)+ATRIB(8),
1946                     ATRIB(7)=ATRIB(7)+ATRIB(8),
1947                     ATRIB(18)=USERF(51),
1948                     ATRIB(8)=TNOW,1;
1949                    ACT,,USERF(37).GE.2,T2F4;     A/C 2 BROKEN
1950                    ACT,.0001,,T2S2;              A/C 2 OK
1951            ;
1952             T2F4 ASSIGN,II=ATRIB(46),
1953                     XX(II)=XX(II)+4,1;
1954                    ACT,.0001,,T2S2;
1955            ;
1956             T2S1 ASSIGN,II=ATRIB(46),1;
1957                    ACT,,XX(II).EQ.0,T02;
1958                    ACT,,XX(II).EQ.4,T2FU;
```

```
1959                    ACT,,XX(II).EQ.2.OR.
1960                       XX(II).EQ.6,T2LO;
1961            ;
1962             T2S2 ASSIGN,II=ATRIB(46),1;
1963                    ACT,,XX(II).EQ.4.OR.
1964                        XX(II).EQ.6,T2LO;
1965                    ACT,,XX(II).EQ.2,T2FU;
1966                    ACT,,XX(II).EQ.0,TO2;
1967            ;
1968             T2FU ASSIGN,ATRIB(13)=7,1;
1969                    ACT,,,DEA6;
1970            ;
```

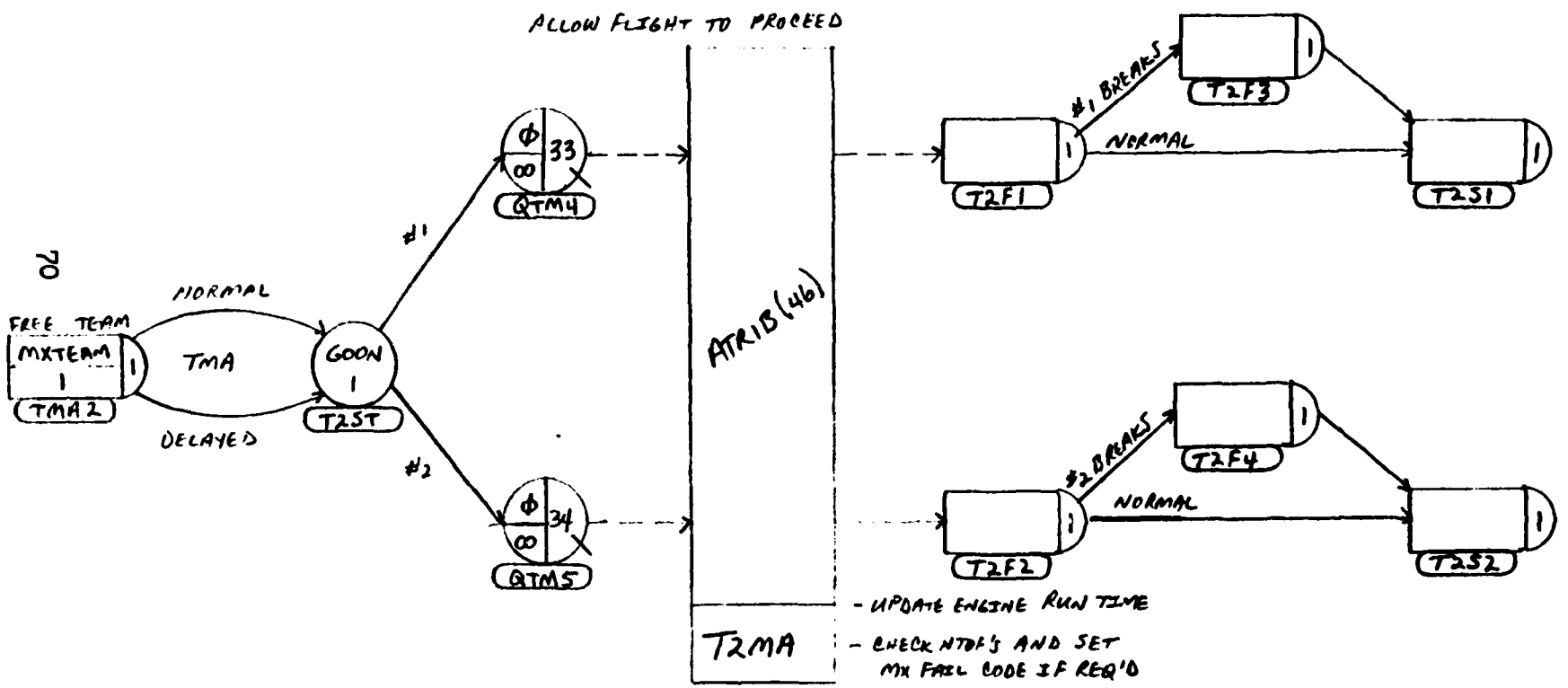


Fig A.10.1 Two-Ship Taxi, Marshall, and Arm

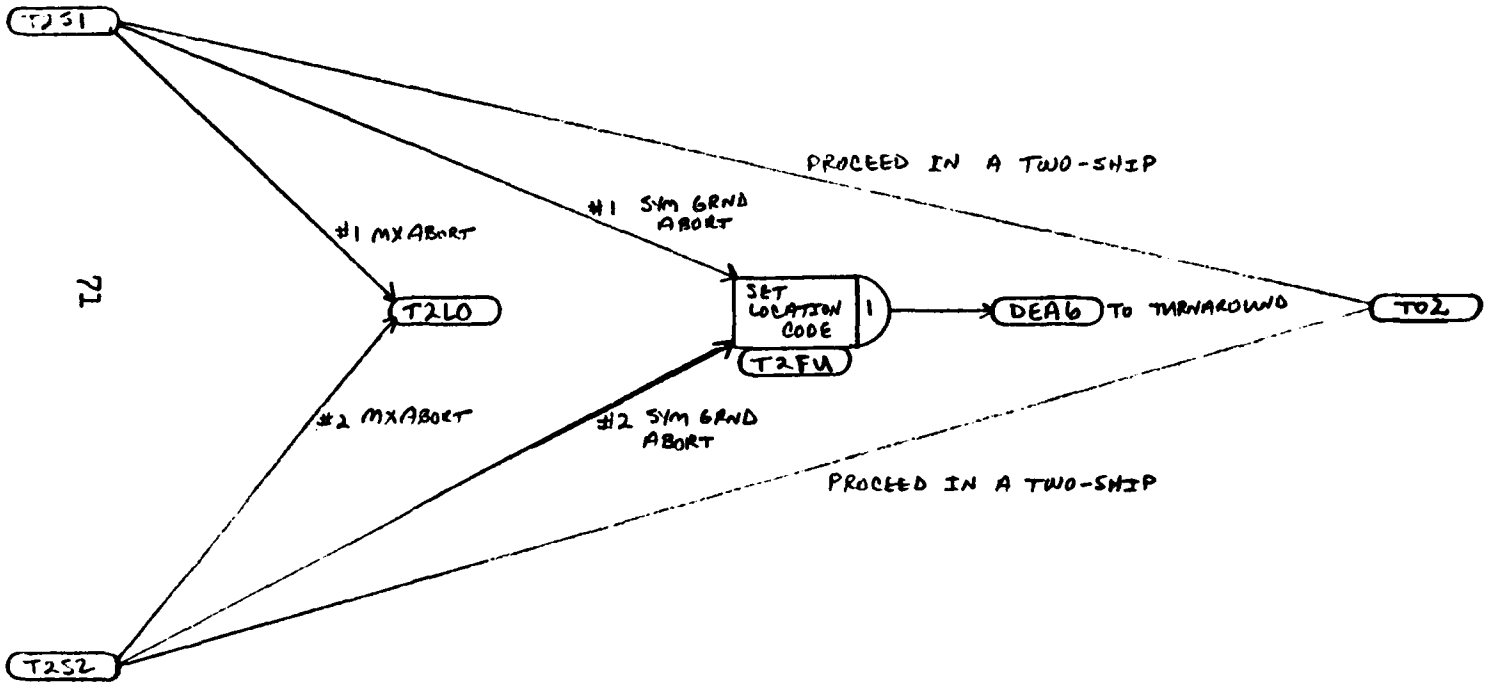


Fig A.10.2 Two-Ship Taxi, Marshall, and Arm

```
1971            ; TWO-SHIP TAKEOFF
1972           ;    THE FLIGHT LEAD ACQUIRES THE RUNWAY FOR THE FLIGHT AND THE
1973           ; FLIGHT TAKES THE ACTIVE AFTER FREEING THEIR ARMING CREWS.
1974           ; RUN-UP CHECKS ARE PERFORMED AND THERE IS A POSSIBILITY OF
1975           ; DELAY.  A/C ARE EVALUATED FOR FAILURES AND ROUTED ACCORD-
1976           ; INGLY.  THE LEAD A/C MAY BE A SYMPATHETIC AIR ABORT, BUT
1977           ; IN ANY CASE IF ONE A/C ABORTS THE OTHER IS SYMPATHETIC.  IF
1978           ; LEAD GROUND ABORTS TWO IS A SYMPATHETIC GROUND ABORT.  THERE
1979           ; IS NO POSSIBILITY OF TWO TAKING OFF IF LEAD ABORTS.
1980           ;
1981            TO2  GOON,1;
1982                   ACT,,ATRIB(45).EQ.1,TOA2;
1983                   ACT,,,TOC2;
1984           ;
1985            TOA2 AWAIT(46),RUNWAY/1,1;
1986                   ACT,.0001,DRAND.GT.XX(68),TOC2;
1987                   ACT,TRIAG(.5,1,2),,TOC2;
1988           ;
1989            TOC2 GOON,1;
1990                   ACT,,ATRIB(45).EQ.1,QTO4;
1991                   ACT,,ATRIB(45).EQ.2,QTO5;
1992           ;
1993            QTO4 QUEUE(38),,,,L2MA;              WAIT TILL FLT IS READY TO TAKEOFF
1994           ;
1995            QTO5 QUEUE(39),,,,L2MA;
1996           ;
1997            L2MA MATCH,46,QTO4/L2F1,QTO5/L2F2;   FLT TAKEOFF
1998           ;
1999            L2F1 ASSIGN,ATRIB(8)=TNOW-ATRIB(8),
2000                    ATRIB(14)=ATRIB(14)+ATRIB(8),
2001                    ATRIB(7)=ATRIB(7)+ATRIB(8),
2002                    ATRIB(18)=USERF(51),
2003                    ATRIB(8)=TNOW,1;
2004                   ACT,,USERF(37).GE.2,L2F3;     A/C 1 BROKEN
2005                   ACT,.0001,,L2S1;              A/C 1 OK
2006           ;
2007            L2F3 ASSIGN,II=ATRIB(46),
2008                    XX(II)=XX(II)+2,1;
2009                   ACT,.0001,,L2S1;
2010           ;
2011            L2F2 ASSIGN,ATRIB(8)=TNOW-ATRIB(8),
2012                    ATRIB(14)=ATRIB(14)+ATRIB(8),
2013                    ATRIB(7)=ATRIB(7)+ATRIB(8),
2014                    ATRIB(18)=USERF(51),
2015                    ATRIB(8)=TNOW,1;
2016                   ACT,,USERF(37).GE.2,L2F4;
2017                   ACT,.0001,,L2S2;
2018           ;
2019            L2F4 ASSIGN,II=ATRIB(46),
2020                    XX(II)=XX(II)+4,1;
```

```
2021                ACT,.0001,,L2S2;
2022          ;
2023           L2S1 ASSIGN,II=ATRIB(46),1;
2024                ACT,1,XX(II).EQ.4,FRS2;
2025                ACT,,XX(II).EQ.0,FRW2;
2026                ACT,1,XX(II).EQ.2.OR.
2027                      XX(II).EQ.6,M2GA;
2028          ;
2029           FRS2 FREE,RUNWAY/1,1;
2030                ACT,,,TOFY;
2031          ;
2032           TOFY ASSIGN,XX(95)=USERF(121),
2033                     XX(94)=XX(94) + 1,
2034                ATRIB(17)=2,XX(95)=USERF(122),1;
2035                ACT,TRIAG(10,15,20),,TOCK;
2036          ;;
2037           TOCK ASSIGN,ATRIB(8)=TNOW-ATRIB(8),
2038                 ATRIB(7)=ATRIB(7)+ATRIB(8),
2039                 ATRIB(18)=USERF(51),
2040                 ATRIB(15)=ATRIB(8),
2041                 ATRIB(8)=TNOW,1;
2042          ;                                   THE CODE ABOVE DOES THE FOLLOWING
2043          ;                                   TIME FLOWN SYM AIR ABORT
2044          ;                                   TOTAL OPERATING TIME
2045          ;                                   UPDATE FAILURE CODE
2046          ;                                   TEMP STORAGE OF FLIGHT TIME
2047          ;                                   RESET ATRIB(8)
2048                ACT,,USERF(13).EQ.1,CRSH;     A/C CRASHES (FAILURE CODE)
2049                ACT,,,APPR;                   A/C RECOVERS TO APPROACH
2050          ;
2051           FRW2 FREE,RUNWAY/1,1;              FLY TWO SHIP
2052                ACT,,,REJ2;
2053          ;
2054           M2GA FREE,RUNWAY/1,1;              MX GRND ABORT A/C 1
2055                ACT,,,DEA3;
2056          ;
2057           L2S2 ASSIGN,II=ATRIB(46),1;
2058                ACT,1,XX(II).EQ.4.OR.
2059                      XX(II).EQ.6,DEA3;       MX GRND ABORT A/C 2
2060                ACT,,XX(II).EQ.0,REJ2;        FLY TWO SHIP
2061                ACT,1,XX(II).EQ.2,L2S3;       SYM GRND ABORT A/C 2
2062          ;
2063           L2S3 ASSIGN,ATRIB(17)=1,1;         SET SYM GRND ABORT CODE
2064                ACT,,,DEA3;
2065          ;
```

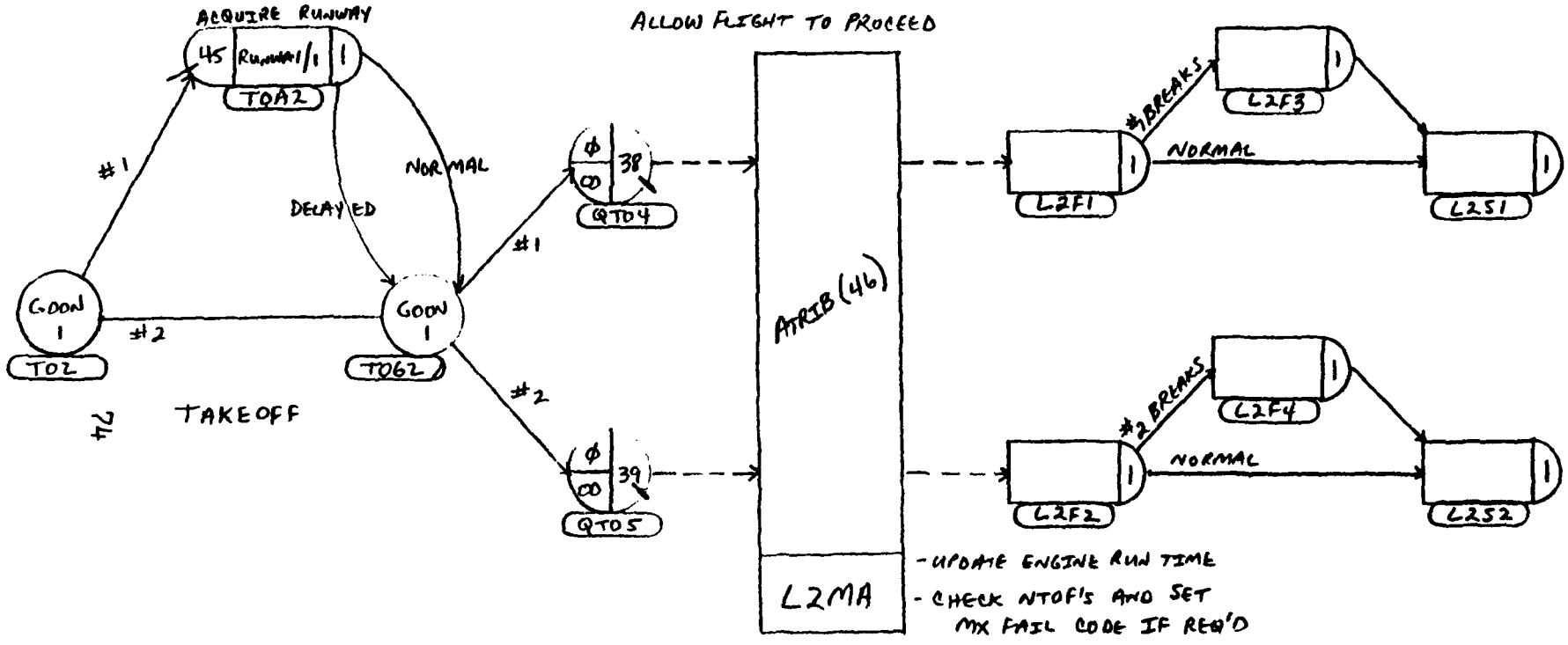


Fig A.11.1 Two-Ship Takeoff

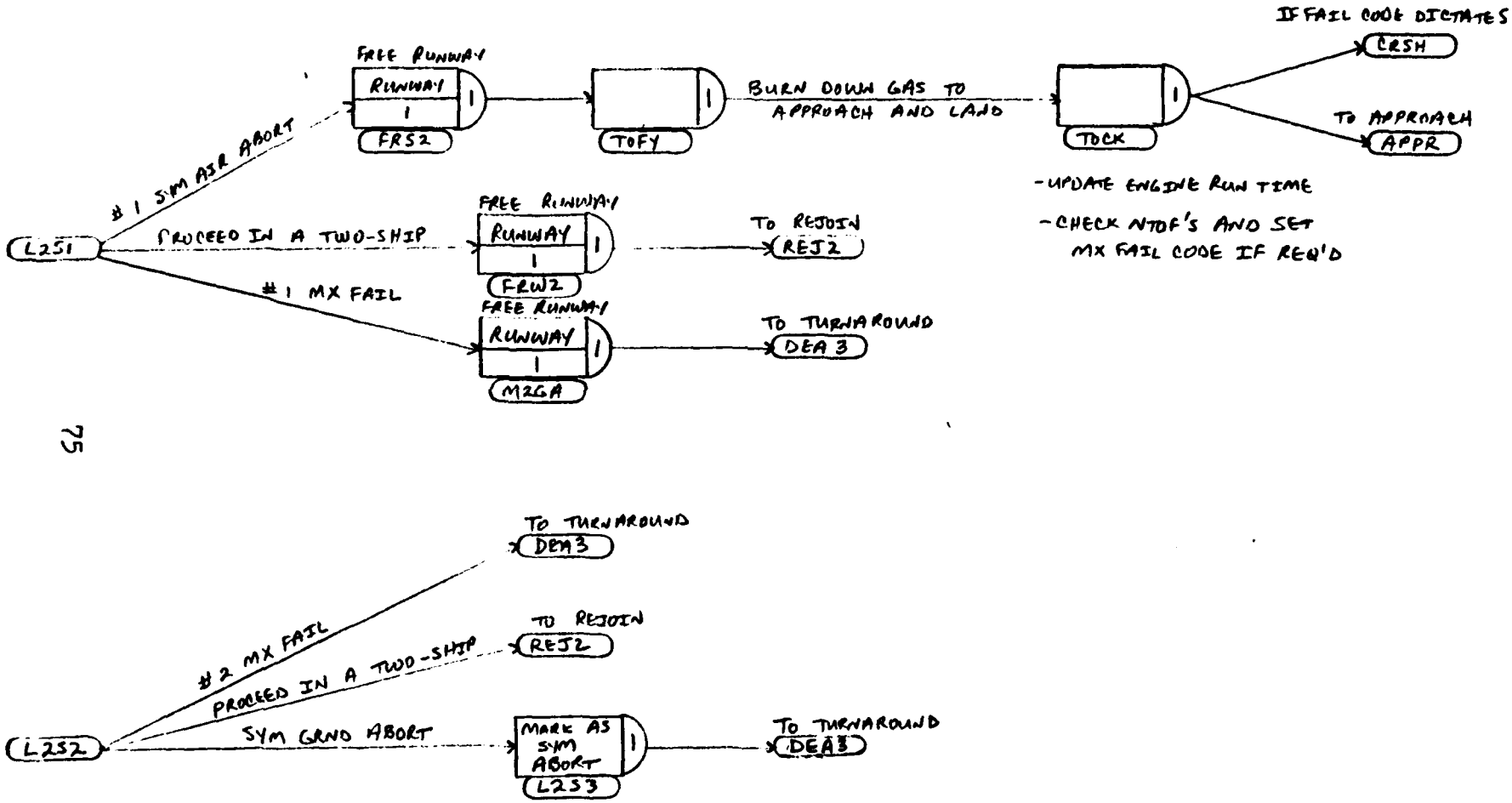


Fig A.11.2 Two-Ship Takeoff

```
2066            ; TWO-SHIP REJOIN
2067           ;    AFTER TAKEOFF THE A/C REJOIN BEFORE PROCEEDING ON THEIR MISSION.
2068           ; A/C ARE EVALUATED FOR FAILURES BEFORE PROCEEDING ON THE MISSION.
2069           ; A/C THAT BREAK ARE ROUTED ACCORDINGLY AND THE OTHER IS A SYM-
2070           ; PATHETIC AIR ABORT.  A/C MAY JETTISON AND/OR BURN DOWN GAS
2071           ; PRIOR TO LANDING, TO GET BELOW MAX GROSS WEIGHT FOR LANDING.  A/C
2072           ; MAY CRASH IF THEIR FAILURE IS SERIOUS ENOUGH.
2073           ;
2074            REJ2 ASSIGN,XX(95)=USERF(121),
2075                        XX(94)=XX(94) + 1,
2076                        XX(95)=USERF(122),1;
2077                 ACT,TRIAG(1,2,3),
2078                  DRAND.GT.XX(69),R2ST;
2079                 ACT,TRIAG(2,3,4),,R2ST;
2080           ;
2081            R2ST GOON,1;
2082                 ACT,,ATRIB(45).EQ.1,QRE4;
2083                 ACT,,ATRIB(45).EQ.2,QRE5;
2084           ;
2085            QRE4 QUEUE(43),,,,R2MA;              WAIT TO REJOIN FLT
2086           ;
2087            QRE5 QUEUE(44),,,,R2MA;
2088           ;
2089            R2MA MATCH,46,QRE4/R2F1,QRE5/R2F2;   REJOIN FLT
2090           ;
2091            R2F1 ASSIGN,ATRIB(8)=TNOW-ATRIB(8),
2092                  ATRIB(15)=ATRIB(8),
2093                  ATRIB(7)=ATRIB(7)+ATRIB(8),
2094                  ATRIB(18)=USERF(51),
2095                  ATRIB(8)=TNOW,1;
2096                 ACT,,USERF(37).GE.2,R2F3;
2097                 ACT,.0001,,R2S1;
2098           ;
2099            R2F3 ASSIGN,II=ATRIB(46),
2100                  XX(II)=XX(II)+2,1;
2101                 ACT,.0001,,R2S1;
2102           ;
2103            R2F2 ASSIGN,ATRIB(8)=TNOW-ATRIB(8),
2104                  ATRIB(15)=ATRIB(8),
2105                  ATRIB(7)=ATRIB(7)+ATRIB(8),
2106                  ATRIB(18)=USERF(51),
2107                  ATRIB(8)=TNOW,1;
2108                 ACT,,USERF(37).GE.2,R2F4;
2109                 ACT,.0001,,R2S2;
2110           ;
2111            R2F4 ASSIGN,II=ATRIB(46),
2112                  XX(II)=XX(II)+4,1;
2113                 ACT,.0001,,R2S2;
2114           ;
2115            R2S1 ASSIGN,II=ATRIB(46),1;
```

```
                    ACT,,XX(II).EQ.2.OR.
                       XX(II).EQ.6,REFY;        MX AIR ABORT
                  ACT,.0001,XX(II).EQ.0,MISS;       FLY TWO SHIP
                  ACT,,XX(II).EQ.4,R2S3;        SYMPATHETIC AIR ABORT
;
 REFY GOON,1;
          ACT,TRIAG(10,15,20),,RECK;
;
 RECK ASSIGN,ATRIB(8)=TNOW-ATRIB(8),
           ATRIB(15)=ATRIB(15)+ATRIB(8),
           ATRIB(7)=ATRIB(7)+ATRIB(8),
           ATRIB(18)=USERF(51),
           ATRIB(8)=TNOW,1;
          ACT,,USERF(13).EQ.1,CRSH;     A/C CRASHED
          ACT,,,APPR;                   A/C TO APPROACH FOR LANDING
;
 R2S3 ASSIGN,ATRIB(17)=2,1;             ASSIGN SYM AIR ABORT CODE
          ACT,,,REFY;
;
 R2S2 ASSIGN,II=ATRIB(46),1;
          ACT,,XX(II).EQ.2,R2S3;        SYMPATHETIC AIR ABORT
          ACT,.0001,XX(II).EQ.0,MISS;   TWO SHIP ON MISSION
          ACT,,XX(II).EQ.4.OR.
               XX(II).EQ.6,REFY;        MX AIR ABORT
;
```

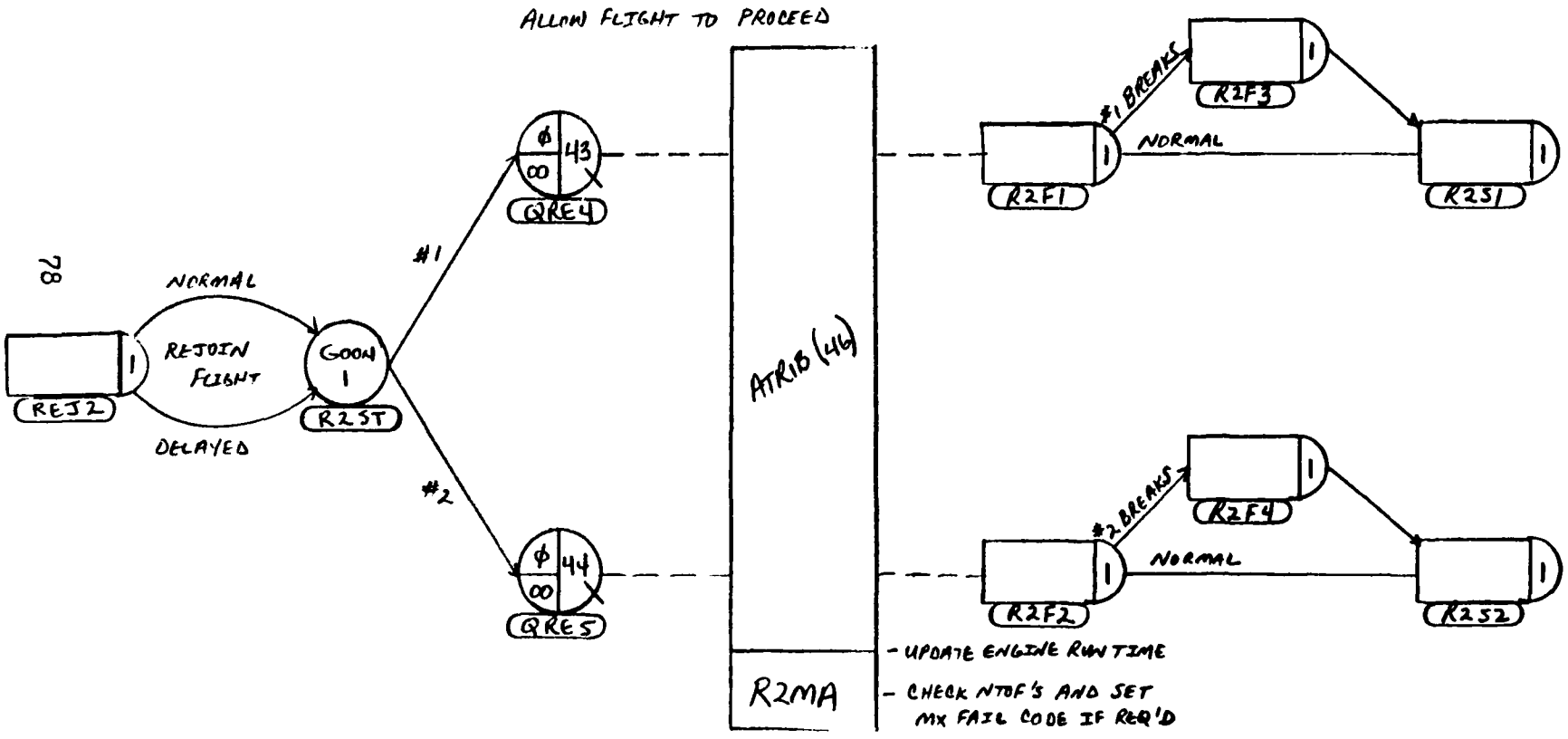


Fig A.12.1 Two-Ship Rejoin

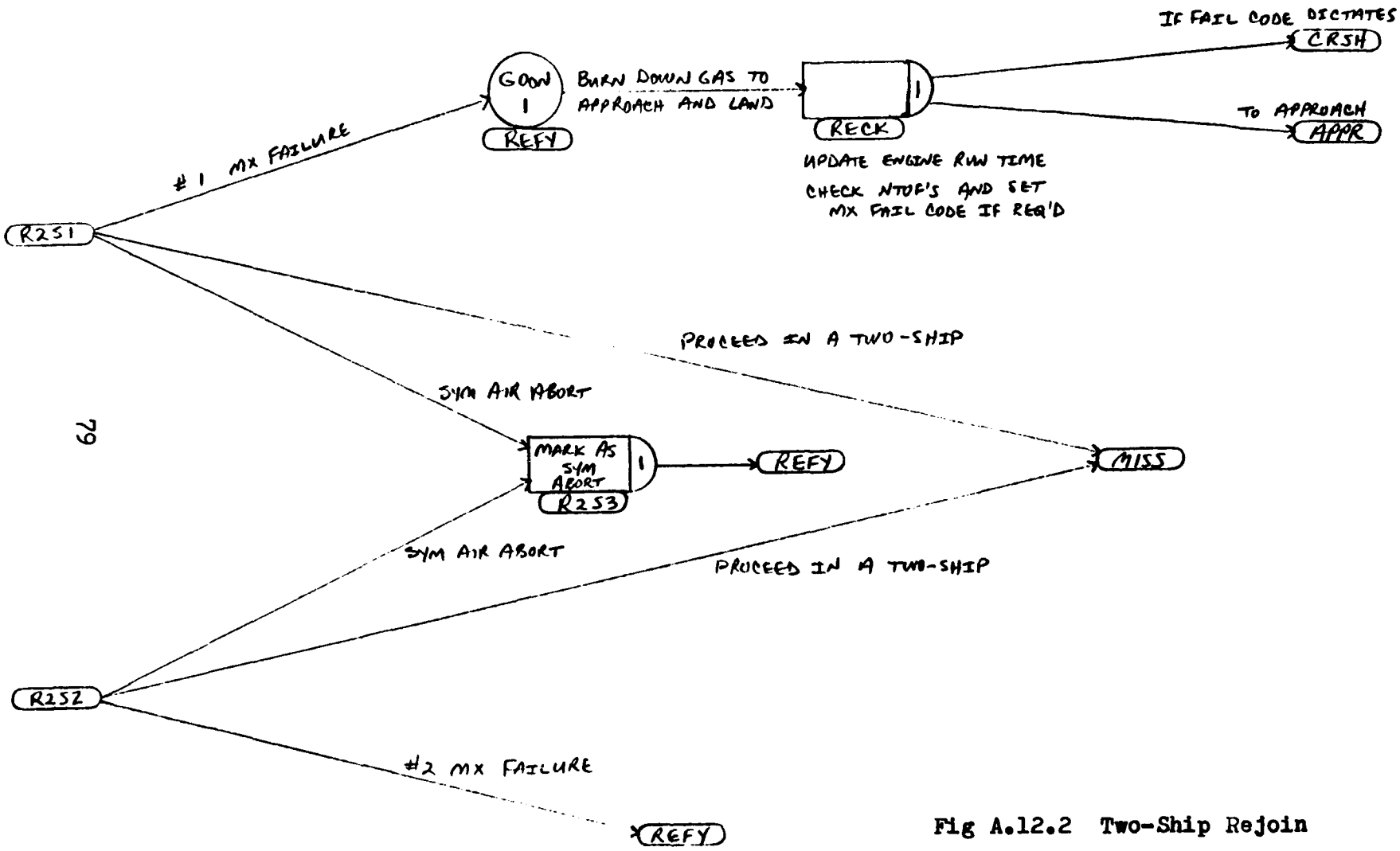


Fig A.12.2 Two-Ship Rejoin

```
2141            ; MISSION
2142           ;    FLIGHTS ARE GIVEN THEIR MISSION DURATION IN THIS SECTION.  A/C
2143           ; CAN BE ATTRITED OR SUFFER BATTLE DAMAGE.  ORDNANCE IS EXPENDED
2144           ; AND EXTERNAL FUEL TANKS ARE JETTISONED IF SPECIFIED CONDITIONS
2145           ; ARE MET.  IF AN A/C IS ATTRITED IT IS ROUTED TO THE JUNK FILE
2146           ; TO BE PRINTED OUT AT THE END OF EACH DAY FOR VALIDATION.  THE
2147           ; SECTION ALSO DETERMINES THE FINAL ORDNANCE CONDITION, WHICH CAN
2148           ; INCLUDE MALFUNCTIONS WHICH CAUSE DELAYS IN SERVICE AT DEARM.
2149           ;    AT THE COMPLETION OF THE MISSION THE MAINTENANCE FAILURE CODE
2150           ; IS UPDATED AND EVALUATED IN CONJUNCTION WITH ANY BATTLE DAMAGE
2151           ; WHICH MAY HAVE OCCURRED.  THE A/C MAY CRASH DEPENDING ON THE
2152           ; THE LEVELS OF FAILURES AND THE PARTICULAR SYSTEMS WHICH ARE
2153           ; AFFECTED.  IF THE AIRCRAFT IS NOT ATTRITED AND DOES NOT CRASH
2154           ; IT IS ROUTED TO APPROACH (APPR) IN ORDER TO ACQUIRE THE RUNWAY
2155           ; FOR LANDING.
2156           ;
2157            MISS ASSIGN,XX(92)=USERF(15),1;       FLY MISSION
2158                 ACT,,ATRIB(16).EQ.99,CRSH;    A/C CRASHED DUE TO ENEMY ACTION
2159                 ACT,XX(92),,MIS1;
2160           ;
2161            MIS1 ASSIGN,ATRIB(8)=TNOW-ATRIB(8),
2162                  ATRIB(7)=ATRIB(7)+ATRIB(8),
2163                  ATRIB(18)=USERF(51),
2164                  ATRIB(15)=ATRIB(15)+ATRIB(8),
2165                  ATRIB(8)=TNOW,1;             CALCULATE MISSION TIME
2166           ;                                   UPDATE A/C OPERATING TIME
2167           ;                                   UPDATE FAILURE CODE
2168           ;                                   UPDATE AIRBORNE TIME
2169           ;                                   RESET TO TNOW
2170                 ACT,,USERF(14).EQ.1,CRSH;     A/C CRASHED DUE TO MX FAILURE
2171                 ACT,,,APPR;                   TO APPROACH
2172           ;
2173            APPR AWAIT(47),RUNWAY/1,1;           CLEARANCE TO LAND ONE A/C
2174                 ACT,.5,ATRIB(39).NE.0,CL03;
2175                 ACT,.5,ATRIB(38).NE.0,CL02;
2176                 ACT,.5,,CL01;
2177            CL03 COLCT,INTVL(39),PILOTFLYTIME03,,1;
2178                 ACT,,,LAND;
2179            CL02 COLCT,INTVL(38),PILOTFLYTIME02,,1;
2180                 ACT,,,LAND;
2181            CL01 COLCT,INTVL(37),PILOTFLYTIME01,,1;
2182                 ACT,,,LAND;
2183           ;
2184            CRSH ASSIGN,ATRIB(2)=USERF(19),1;     DELETES FROM CURRENT INVENTORY
2185                 ACT,,,CRS2;
2186           ;
2187            CRS2 AWAIT(99),JUNK,1;               FILE STORING CRASHED A/C FOR
2188           ;                                       STATISTICS FROM DAILY CLEAN UP
2189           ;                                       ROUTINE
2190                TERMINATE;
```

2191

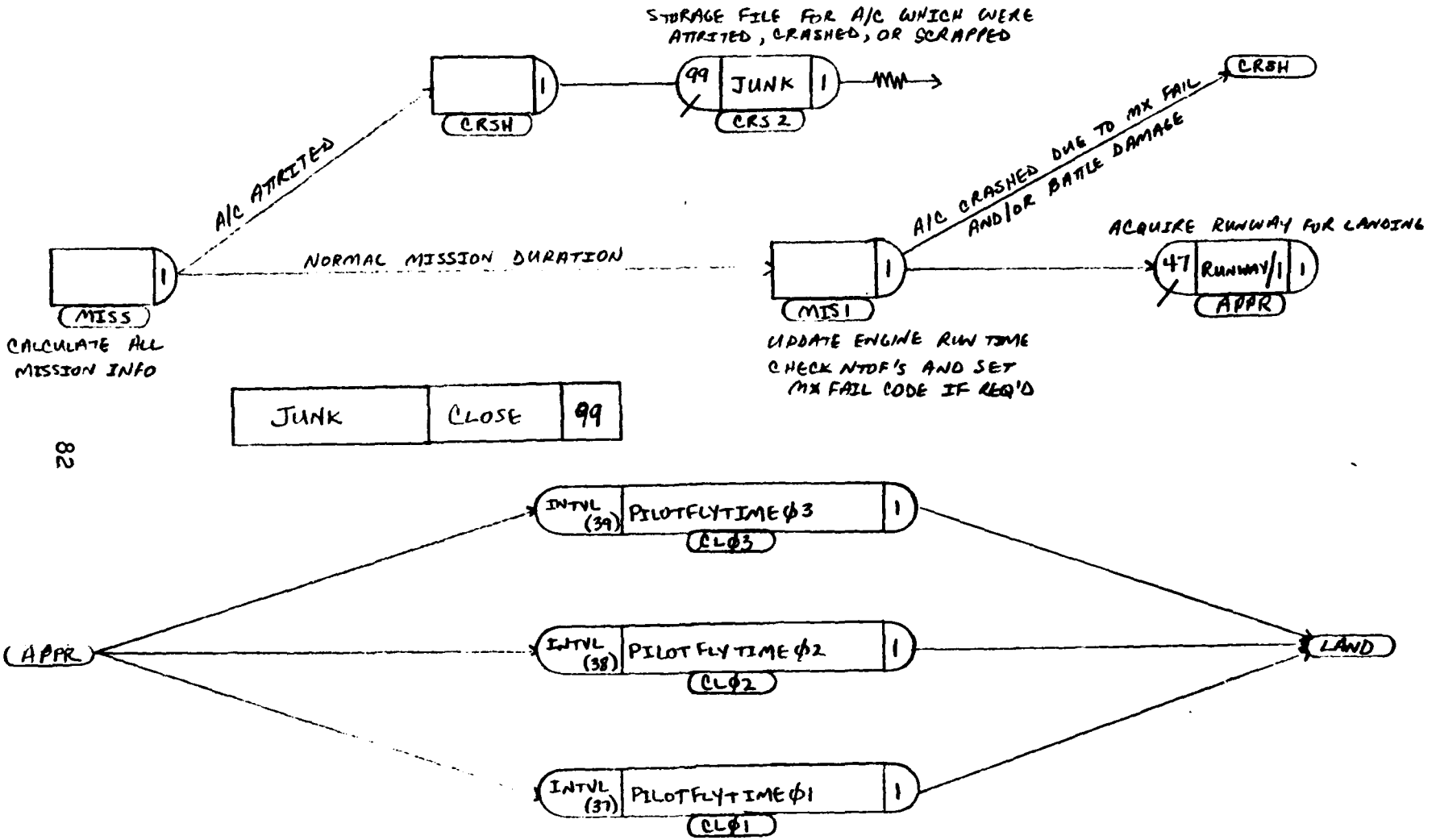


Fig A.13.1 Mission and Approach

```
2192            ; LANDING, ROLLOUT, TAXI TO DEARM, DEARM
2193           ;    AFTER THE A/C HAS LANDED AND ROLLED OUT A DETERMINATION IS MADE
2194           ; WHETHER IT CAN TAXI OR IF IT MUST BE TOWED.  IF IT MUST BE TOW-
2195           ; ED THE PILOT IS SEPARATED AND RETURNED TO HIS SQUADRON AREA.
2196           ; FOR EITHER CASE THE A/C GOES TO THE DEARMING AREA WHERE IT
2197           ; ACQUIRES A DEARMING CREW.  IF THE A/C WAS TOWED AND/OR IT HAD
2198           ; A MAINTENANCE PROBLEM, IT IS ROUTED TO MAINTENANCE FOR REPAIR.
2199           ; A/C THAT CAN TAXI PROCEED TO THEIR SQUADRON AREA AND HOTPIT
2200           ; REFUEL ON THE WAY IF IT IS CONVENIENT FOR THEIR TAXI ROUTE.
2201           ; A/C WHICH HAVE ORDNANCE MALFUNCTIONS ASSIGNED AT MISSION EXPER-
2202           ; IENCE LONGER SERVICE TIMES.
2203           ;
2204            LAND ASSIGN,ATRIB(8)=TNOW-ATRIB(8),
2205                    ATRIB(7)=ATRIB(7)+ATRIB(8),
2206                    ATRIB(18)=USERF(51),
2207                    ATRIB(15)=ATRIB(15)+ATRIB(8),
2208                    ATRIB(8)=TNOW,ATRIB(13)=12,
2209                    XX(95)=USERF(124),
2210                    ATRIB(14)=0,2;              ATRIB(13) IS LOCATION CODE
2211                   ACT,TRIAG(5,10,12),
2212                       USERF(12).EQ.1,MSFR;     A/C BROKE ON RNWY,TOW TO DEARM
2213                   ACT,USERF(67),
2214                       USERF(12).EQ.1,PSEP;     FREE PILOT FROM TOWED A/C
2215                   ACT,1,,LAN1;                 A/C CLEARS RUNWAY NORMALLY
2216           ;
2217            MSFR FREE,RUNWAY/1,1;               A/C HAS CLEARED RUNWAY
2218                   ACT,,,DEAR;
2219           ;
2220            DEAR AWAIT(49),DEARM/1,1;           WAIT FOR DEARM CREW
2221                   ACT/47,USERF(75),,DEA1;
2222           ;
2223            DEA1 FREE,DEARM/1,1;                FREE DEARM CREW
2224                   ACT,,,DEA2;
2225           ;
2226            DEA2 ASSIGN,ATRIB(32)=0,ATRIB(13)=10,1;A/C W/O PILOT FROM DEARM AREA
2227                   ACT,,,MAIN;
2228           ;
2229            LAN1 ASSIGN,ATRIB(8)=TNOW-ATRIB(8),
2230                   ATRIB(7)=ATRIB(7)+ATRIB(8),
2231                   ATRIB(18)=USERF(51),
2232                   ATRIB(14)=ATRIB(14)+ATRIB(8),
2233                   ATRIB(8)=TNOW,ATRIB(13)=10,1; AT(14) IS GROUND RUN TIME
2234           ;                                    AT(13) IS LOCATION CODE
2235                   ACT,,,MSF1;                  A/C CLEARS RUNWAY
2236           ;
2237            MSF1 ASSIGN,XX(95)=USERF(124),1;
2238                   ACT,,,MSF2;
2239           ;
2240            MSF2 FREE,RUNWAY/1,2;
2241                   ACT,,USERF(12).EQ.1,DEAR;    A/C BROKE,NEEDS TOW FROM DEARM
```

```
2242                ACT,,USERF(12).EQ.1,PSEP;     FREE PILOT FROM TOWED A/C
2243              ACT,,,DEA3;                     A/C OK
2244            ;
2245            DEA3 AWAIT(48),DEARM/1,1;         WAIT FOR DEARM CREW
2246                 ACT/48,USERF(75),,DEA4;
2247            ;
2248            DEA4 FREE,DEARM/1,1;              FREE DEARM CREW
2249                 ACT,,,DEA5;
2250            ;
2251            DEA5 ASSIGN,ATRIB(8)=TNOW-ATRIB(8),
2252                  ATRIB(7)=ATRIB(7)+ATRIB(8),
2253                  ATRIB(18)=USERF(51),
2254                  ATRIB(14)=ATRIB(14)+ATRIB(8),
2255                  XX(95)=USERF(124),
2256                  ATRIB(8)=TNOW,ATRIB(13)=10,2;
2257                 ACT,,USERF(37).GE.2.AND.
2258                      USERF(12).EQ.1.OR.
2259                      ATRIB(16).GE.2.AND.
2260                      USERF(12).EQ.1,DEA7;    A/C BROKE/DAMAGED,NEEDS TOW
2261                 ACT,,USERF(37).GE.2.AND.
2262                      USERF(12).EQ.0.OR.
2263                      ATRIB(16).GE.2.AND.
2264                      USERF(12).EQ.0,MAIN;    A/C BROKE/DAMAGED, CAN TAXI
2265                 ACT,USERF(67),
2266                      USERF(37).GE.2.AND.
2267                      USERF(12).EQ.1.OR.
2268                      ATRIB(16).GE.2.AND.
2269                      USERF(12).EQ.1,PSEP;    FREE PILOT FROM TOWED A/C
2270                 ACT,,USERF(37).LT.2.AND.
2271                      ATRIB(16).LT.2,DEA6;    A/C NORMAL TAXI
2272            ;
2273            DEA7 ASSIGN,ATRIB(32)=0,1;        NO PILOT IN TOWED A/C
2274                 ACT,,,MAIN;
2275            ;
2276            DEA6 ASSIGN,ATRIB(3)=USERF(21),1; GET SQ PARKING SPOT
2277                 ACT,USERF(64),
2278                     USERF(77).EQ.1,HOT;      NON-SHELTERED A/C FROM A SQ
2279            ;                                 CONVENIENTLY LOCATED TAXI TO HOT
2280            ;
2281                 ACT,USERF(61),,SQPA;         SQ NOT CONVENIENT TO HOTPIT
2282            ;                                 PROCEED TO SQ PARKING
2283            ;
```

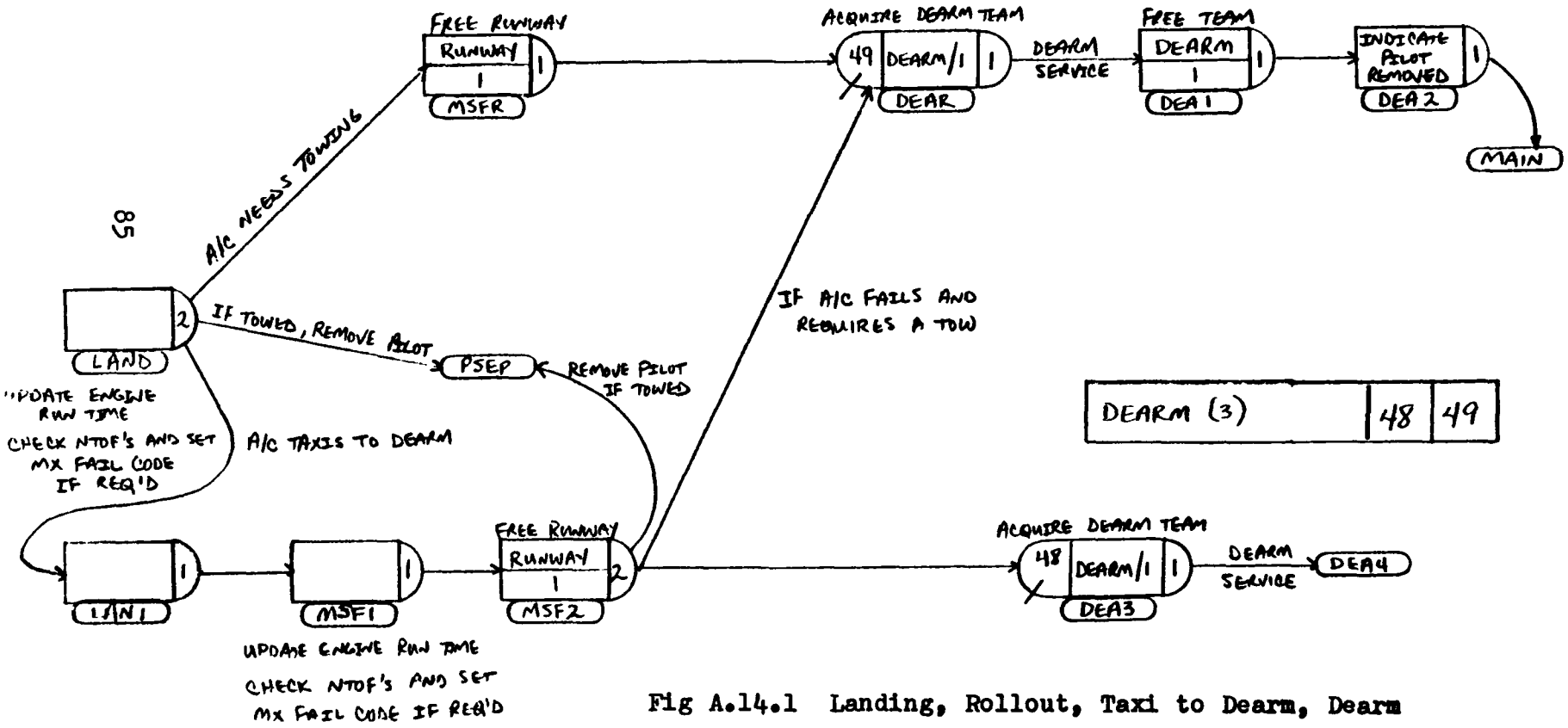


Fig A.14.1 Landing, Rollout, Taxi to Dearm, Dearm

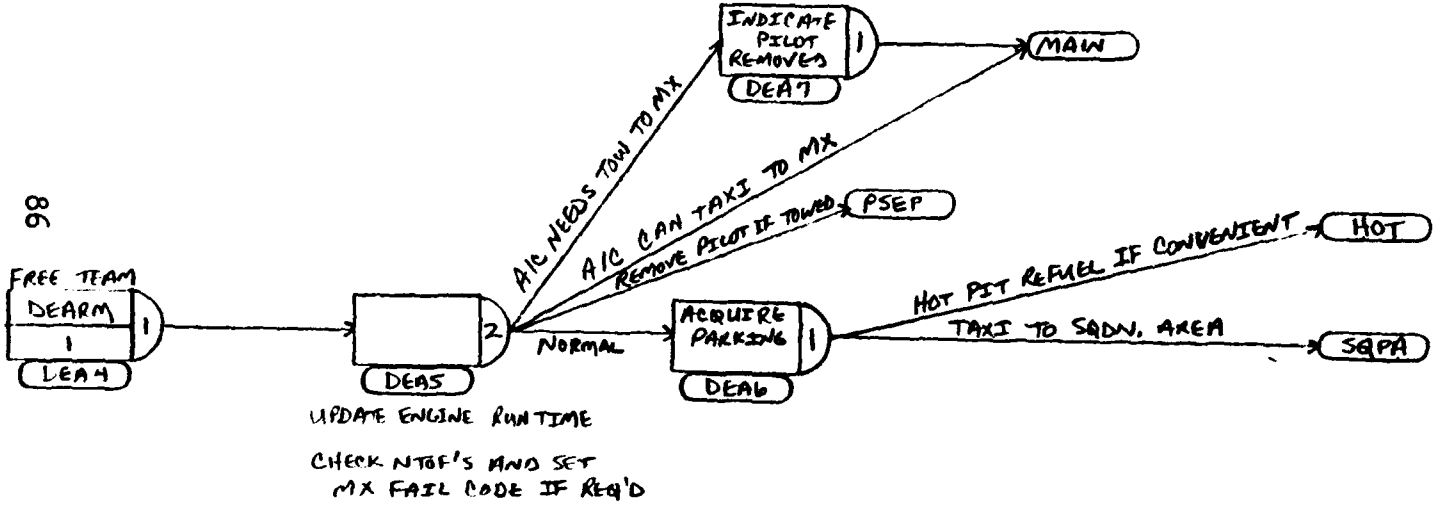


Fig A.14.2 Landing, Rollout, Taxi to Dearm, Dearm

```
2284            ; HOTPIT REFUELING
2285           ;    THOSE A/C WHOSE SQUADRON AREA IS CONVENIENTLY LOCATED ON THE
2286           ; TAXI ROUTE OF A/C RETURNING TO THEIR AREAS REFUEL AT HOTPIT IF
2287           ; FUEL IS AVAILABLE AND A HOTPIT IS AVAILABLE.  IF THE A/C BREAKS
2288           ; TAXIING TO OR DURING HOTPIT REFUELING, THE A/C WILL TAXI OR BE
2289           ; TOWED TO MAINTENANCE.  IF IT IS TOWED THE PILOT IS TAKEN OUT
2290           ; OF THE A/C AND ROUTED TO HIS SQUADRON AREA.  A/C PROCEED TO
2291           ; THEIR SQUADRON AREA AFTER COMPLETING HOTPIT REFUELING.
2292           ;
2293            HOT  ASSIGN,ATRIB(8)=TNOW-ATRIB(8),
2294                        ATRIB(7)=ATRIB(7)+ATRIB(8),
2295                        ATRIB(14)=ATRIB(14)+ATRIB(8),
2296                        ATRIB(18)=USERF(51),
2297                        XX(95)=USERF(124),
2298                        ATRIB(8)=TNOW,ATRIB(13)=11,2;UPDATE TIME
2299                       ACT,USERF(61),USERF(37).LT.2.AND.
2300                           NNQ(50).GT.0.OR.
2301                           USERF(37).LT.2.AND.
2302                           XX(60).LE.0,SQPA;        BALK FROM HOTPIT, A/C IN LINE
2303                       ACT,,USERF(37).GE.2.AND.
2304                            USERF(12).EQ.0,HOT6;    MX PROB,NO HOT PIT,TAXI OK
2305                       ACT,,USERF(37).GE.2.AND.
2306                            USERF(12).EQ.1,HOT5;    A/C BROKE - NEEDS TOW
2307                       ACT,USERF(67),
2308                            USERF(37).GE.2.AND.
2309                            USERF(12).EQ.1,PSEP;    FREE PILOT FROM TOWED A/C
2310                       ACT,,USERF(37).LT.2.AND.
2311                            NNQ(50).EQ.0.AND.
2312                            XX(60).GT.0,HOTP;       GO TO HOTPIT REFUEL
2313           ;
2314            HOTP AWAIT(50),HOTPIT/1,1;              AWAIT A FREE PIT
2315                  ACT,,,HOT1;
2316           ;
2317            HOT1 ASSIGN,ATRIB(15)=USERF(71),
2318                        XX(60)=XX(60)-ATRIB(15),1;  SET FUEL REQUIRED
2319                  ACT/4,USERF(78),,HOT3;            HYDRANT REFUELING SERVICE
2320           ;
2321            HOT3 FREE,HOTPIT/1,1;                   FREE PIT FOR NEXT A/C
2322                  ACT,,,HOT4;
2323           ;
2324            HOT4 ASSIGN,ATRIB(8)=TNOW-ATRIB(8),
2325                        ATRIB(7)=ATRIB(7)+ATRIB(8),
2326                        ATRIB(18)=USERF(51),
2327                        ATRIB(8)=TNOW,ATRIB(14)=0,
2328                        XX(95)=USERF(124),
2329                        ATRIB(15)=0,1;              CHK FOR FAILURE, RESET TIME, FUEL
2330                       ACT,USERF(67),
2331                           USERF(37).GE.2.AND.
2332                           USERF(12).EQ.1,PSEP;     FREE PILOT FROM TOWED A/C
2333                       ACT,,USERF(37).GE.2.AND.
```

```
 2334                    USERF(12).EQ.1,HOT5;      A/C BROKE - NEEDS TOW
2335              ACT,,USERF(37).GE.2.AND.
2336                   USERF(12).EQ.0,HOT6;     A/C BROKE BUT OK TO TAXI
2337              ACT,USERF(61),,SQPA;           A/C TO SQ PARKING
2338         ;
2339          HOT5 ASSIGN,ATRIB(32)=0,1;         NO PILOT IN TOWED A/C
2340              ACT,,,HOT6;
2341         ;
2342          HOT6 ASSIGN,ATRIB(3)=USERF(22),1;  GIVE UP SQ PARKING, GO TO MAIN
2343              ACT,,,MAIN;
2344         ;
```

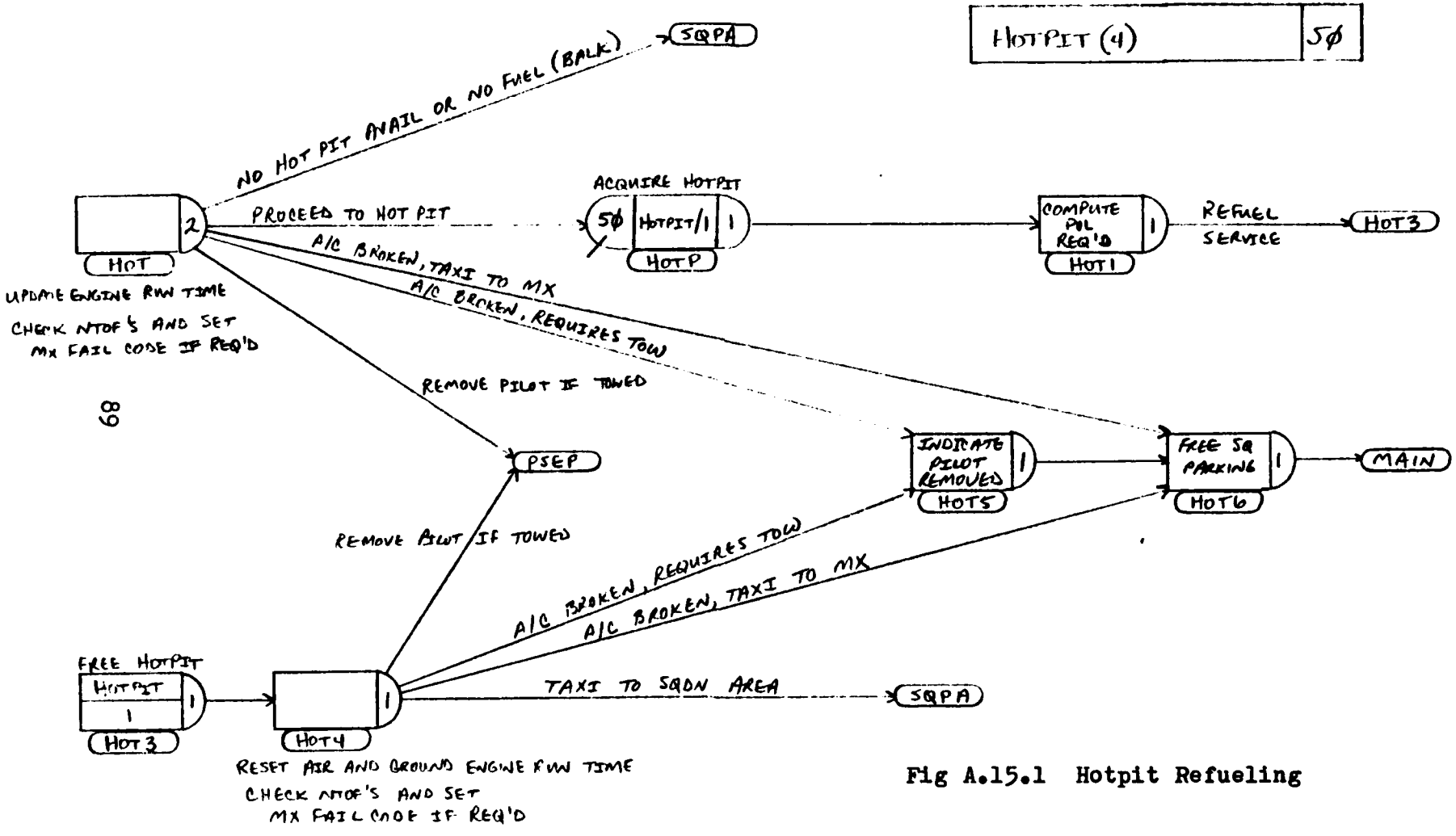


Fig A.15.1 Hotpit Refueling

```
2345            ; ENGINE SHUT DOWN
2346           ;    WHEN AN A/C REACHES ITS SQUADRON AREA IT IS PARKED IN ITS
2347           ; ASSIGNED SPACE AND THE ENGINE IS SHUT DOWN.  THE PILOT LEAVES THE
2348           ; A/C (PSEP).  IF THE A/C EXPERIENCED A FAILURE ENROUTE TO THE
2349           ; SQUADRON AREA IT IS SCHEDULED FOR MAINTENANCE (SQ LEVEL
2350           ; OR MMT).  A/C THAT ARE TURNABLE ACQUIRE A CREW CHIEF AND BEGIN
2351           ; TURNAROUND SERVICING.  THE A/C STAYS IN THE SQUADRON AREA FOR
2352           ; MAINTENANCE.
2353           ;    AFTER ACQUIRING A CREW CHIEF A DETERMINATION IS MADE WHETHER THE
2354           ; A/C REQUIRES RECONFIGURATION.  IF IT DOES, THE A/C IS RECON-
2355           ; FIGURED AND PROCEEDS TO TURNAROUND PROCESSING IN THE PARALLEL
2356           ; SERVICE OPERATION (PSO).  IF NO RECONFIGURATION IS REQUIRED
2357           ; THE A/C PROCEEDS DIRECTLY TO PSO.
2358           ;
2359            SQFA ASSIGN,ATRIB(8)=TNOW-ATRIB(8),
2360                    ATRIB(7)=ATRIB(7)+ATRIB(8),
2361                    ATRIB(14)=ATRIB(14)+ATRIB(8),
2362                    XX(95)=USERF(124),
2363                    ATRIB(18)=USERF(51),2;
2364                   ACT,,,PSEP;                  SEPARATE PILOT AT ENG. SHUTDOWN
2365                   ACT,,USERF(37).GE.2,SPMX;    TO SQ MAINT -SEE IF MMT REQ'D
2366                   ACT,,,MXTM;                  AWAIT CREW CHIEF ASSIGNMENT
2367           ;
2368            PSEP ASSIGN,ATRIB(8)=ATRIB(38),
2369                       ATRIB(9)=ATRIB(39),ATRIB(10)=ATRIB(40),
2370                       ATRIB(11)=ATRIB(41),1;
2371           ;
2372            PSE1 ASSIGN,ATRIB(1)=ATRIB(31),
2373                    ATRIB(2)=ATRIB(32),ATRIB(3)=ATRIB(33),
2374                    ATRIB(4)=ATRIB(34),ATRIB(5)=ATRIB(35),
2375                    ATRIB(6)=ATRIB(36),ATRIB(7)=ATRIB(37),
2376                    ATRIB(12)=ATRIB(42),
2377                    XX(95)=USERF(123),1;         RECREATE PILOT
2378                   ACT,,ATRIB(32).EQ.0,PSE2;     NO PILOT WAS IN THE A/C
2379                   ACT,10,ATRIB(1).EQ.1,PL1;     SMOKE & A COKE, BACK TO SQ RDYPOOL
2380                   ACT,10,ATRIB(1).EQ.2,PL2;
2381                   ACT,10,ATRIB(1).EQ.3,PL3;
2382                   ACT,10,ATRIB(1).EQ.4,PL4;
2383                   ACT,10,ATRIB(1).EQ.5,PL5;
2384                   ACT,10,ATRIB(1).EQ.6,PL6;
2385           ;
2386            PSE2 TERMINATE;
2387           ;
2388           ;
2389            MXTM AWAIT(51),MXTEAM/1,1;          AWAIT CREW CHIEF
2390                   ACT,,ATRIB(17).EQ.1,PSO;     A/C WAS SYM GRND ABORT, TO PSO
2391                   ACT,TRIAG(3,4,5),,MXT1;      MX POST-FLIGHT ACTIVITY
2392           ;
2393            MXT1 ASSIGN,ATRIB(8)=USERF(73),
2394                       XX(95)=USERF(125),
```

```
2395                    ATRIB(15)=USERF(71),1;    DETERMINE CONFIGURATION REQ'D
2396                ACT,TRIAG(3,4,5),
2397                   ATRIB(8).EQ.ATRIB(12),PSO;NO RECONFIG REQ'D.
2398                ACT,TRIAG(3,4,5),,RECO;       MX POST-FLIGHT ACTIVITY
2399            ;
2400             RECO ASSIGN,ATRIB(15)=USERF(72),
2401                    ATRIB(12)=ATRIB(8),1;     RECALC FUEL REQ'D, CONFIG SET
2402                ACT/1,USERF(76),,PSO;         RECONFIG SERVICE
2403            ;
```

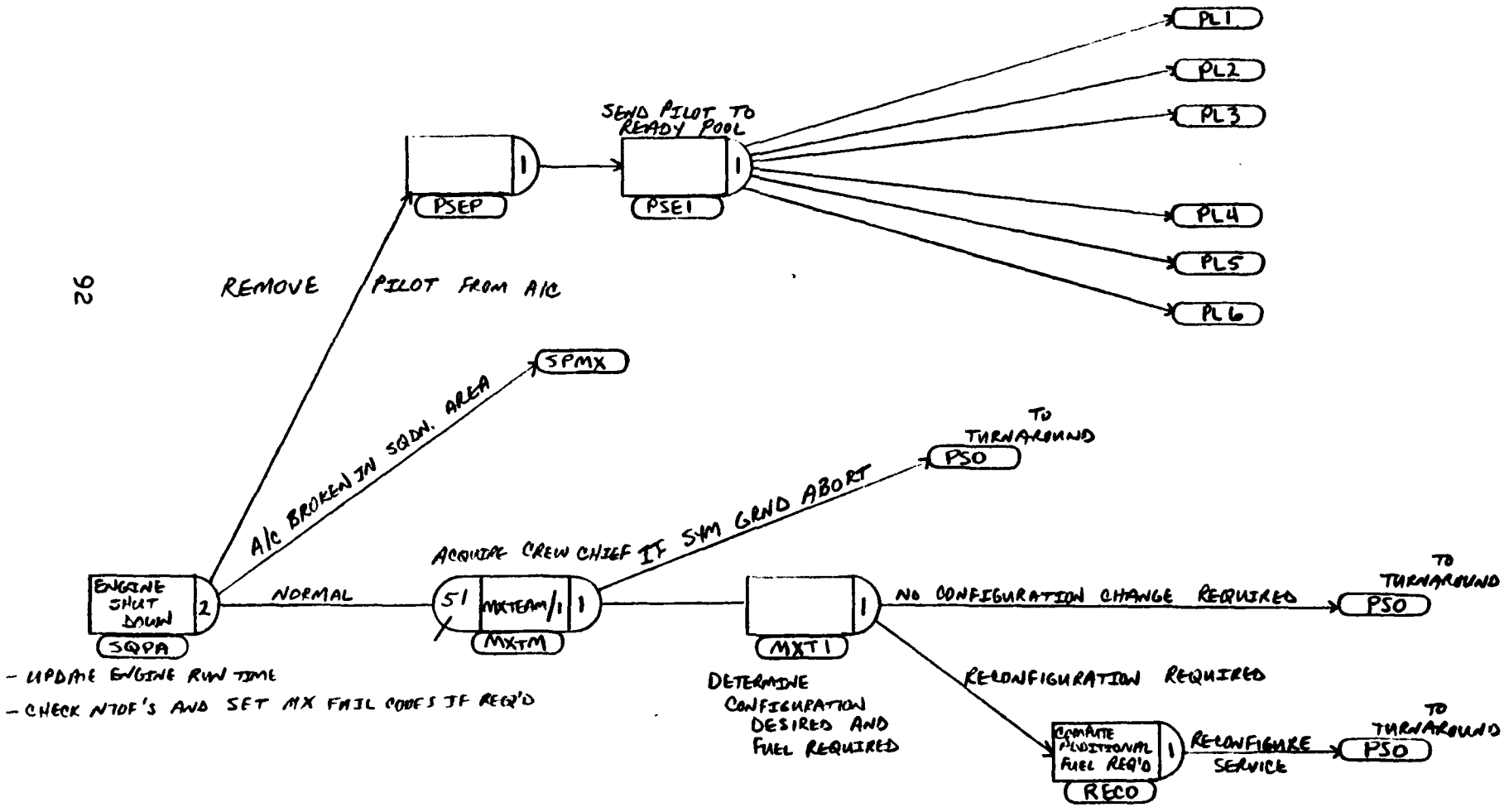


Fig A.16.1 Engine Shut Down

```
2404            ; TURNAROUND SERVICE OPERATION
2405           ;    THE TURNAROUND SERVICE OPERATION IS A PARALLEL SERVICE OPERA-
2406           ; TION IN WHICH THE SERVICES ARE PERFORMED CONCURRENTLY.  THE
2407           ; SERVICES ARE REARMING, MAINTENANCE POST-FLIGHT AND REFUELING,
2408           ; IF REQUIRED.  THE A/C ACQUIRES A REARMING TEAM, RECEIVES ITS
2409           ; ORDNANCE AND THEN FREES THE TEAM.  IF THE AIRCRAFT WAS A SYMPA-
2410           ; THETIC ABORT AND ALREADY HAS ORDNANCE A PATH IS AVAILABLE
2411           ; AROUND REARMING.  AT THE SAME TIME THE CREW CHIEF PERFORMS THE
2412           ; MAINTENANCE POST-FLIGHT, THE AIRCRAFT IS REFUELED, IF REQUIRED.
2413           ; IF THE A/C IS PARKED IN A SHELTER IT IS REFUELED THERE.  IF THE
2414           ; AIRCRAFT IS NOT SHELTERED IT REQUIRES A FUEL TRUCK.  IF NO FUEL
2415           ; IS AVAILABLE AND THE A/C REQUIRES FUEL, IT WAITS FOR REFUELING
2416           ; IN THE PARKING SPACE.  WHEN ALL THREE SERVICES ARE COMPLETED
2417           ; THE CREW CHIEF IS RELEASED AND THE AIRCRAFT IS REASSEMBLED AT
2418           ; THE MATCH NODE AND IT IS READY TO GO THROUGH THE STATISTICAL
2419           ; ROUTINE TO THE AIRCRAFT READY POOL FOR ITS SQUADRON.
2420           ;
2421            PSO  GOON,3;                          TURNAROUND SERVICING ROUTINE
2422                   ACT,,ATRIB(9).NE.1.OR.
2423                        ATRIB(10).NE.1,REAR;      TO REARM, IF REQ'D
2424                   ACT,,ATRIB(9).EQ.1.AND.
2425                        ATRIB(10).EQ.1,REA2;      IF ARMED, BRANCH AROUND REARM SERVICE
2426                   ACT/3,TRIAG(4,5,6),,MXPF;      MX POST-FLIGHT
2427                   ACT,,ATRIB(15).EQ.0,REF3;      A/C HOT PIT REFUELED, SKIP REFUE
2428                   ACT,,ATRIB(15).NE.0.AND.
2429                        XX(60).GT.0,GREF;         A/C NEEDS GAS AND GAS AVAILABLE
2430                   ACT,,ATRIB(15).NE.0.AND.
2431                        XX(60).LE.0,REF7;         NEEDS GAS, NONE AVAIL, CLOSE GATE
2432           ;
2433            REAR AWAIT(52),REARM/1,1;             WAIT FOR REARM CREW
2434                   ACT/2,USERF(76)+3,,REA1;       REARM SERVICE
2435           ;
2436            REA1 FREE,REARM/1,1;                  RELEASE ARMING CREW
2437                   ACT,,,REA2;
2438           ;
2439            REA2 QUEUE(55),,,,PSMA;               Q BEFORE MATCH, END CONCURRENT
2440           ;                                        TURNAROUND SERVICE
2441           ;
2442            MXPF QUEUE(56),,,,PSMA;               Q BEFORE MATCH
2443           ;
2444            GREF GOON,1;
2445                   ACT,,ATRIB(3).EQ.1,REF1;       REFUEL IN A SHELTER
2446                   ACT,,,REFU;                    REFUELED BY A TRUCK
2447           ;
2448            REF1 ASSIGN,XX(60)=XX(60)-ATRIB(15),1;REDUCE POL BY AMOUNT USED
2449                   ACT/5,USERF(78),,REF2;         SHELTER REFUEL SERVICE
2450           ;
2451            REF2 ASSIGN,ATRIB(14)=0,ATRIB(15)=0,1;RESET TIME KEEPERS
2452                   ACT,,,REF3;
2453           ;
```

```
 2454            REF3 QUEUE(57),,,,PSMA;              Q BEFORE MATCH
2455            ;
2456             REFU AWAIT(53),REFUEL/1,1;           WAIT FOR FUEL TRUCK
2457            ;
2458             REF4 ASSIGN,XX(60)=XX(60)-ATRIB(15),1;REDUCE POL BY AMOUNT REQ'D
2459                    ACT/6,TRIAG(5,8,12)+3,,REF5;  TRUCK REFUELING SERVICE
2460            ;
2461             REF5 FREE,REFUEL/1,1;                RELEASE FUEL TRUCK
2462                    ACT,,,REF2;
2463            ;
2464             REF7 CLOSE,FUELAVAL,1;               NO FUEL AVAILABLE, CLOSE GATE
2465                    ACT,,,REF8;
2466            ;
2467             REF8 AWAIT(54),FUELAVAL,1;           WAIT FOR FUEL
2468                    ACT,,,GREF;
2469            ;
2470             PSMA MATCH,2,REA2/TERM,
2471                     MXPF/TERM,REF3/PSOC;         A/C DONE WITH CONCURRENT SERVICE
2472            ;
2473            ;
2474             PSOC FREE,MITEAM/1,1;                RELEASE CREW CHIEF
2475                    ACT,,ATRIB(27).NE.0,CL09;
2476                    ACT,,ATRIB(26).NE.0,CL08;
2477                    ACT,,,CL07;
2478            ;
2479             CL09 COLCT,INTVL(27),TURNSERVDAY03,,1;
2480                    ACT,,,PSOX;
2481            ;
2482             CL08 COLCT,INTVL(26),TURNSERVDAY02,,1;
2483                    ACT,,,PSOX;
2484            ;
2485             CL07 COLCT,INTVL(25),TURNSERVDAY01,,1;
2486                    ACT,,,PSOX;
2487            ;
2488             PSOX ASSIGN,ATRIB(17)=0,
2489                     ATRIB(13)=ATRIB(1),1;        RESET SYM CODE AND LOCATION CODE
2490                    ACT,,USERF(17).EQ.1,ARP1;     A/C TO THEIR SQ A/C READYPOOL
2491                    ACT,,USERF(17).EQ.2,ARP2;
2492                    ACT,,USERF(17).EQ.3,ARP3;
2493                    ACT,,USERF(17).EQ.4,ARP4;
2494                    ACT,,USERF(17).EQ.5,ARP5;
2495                    ACT,,USERF(17).EQ.6,ARP6;
2496            ;
```

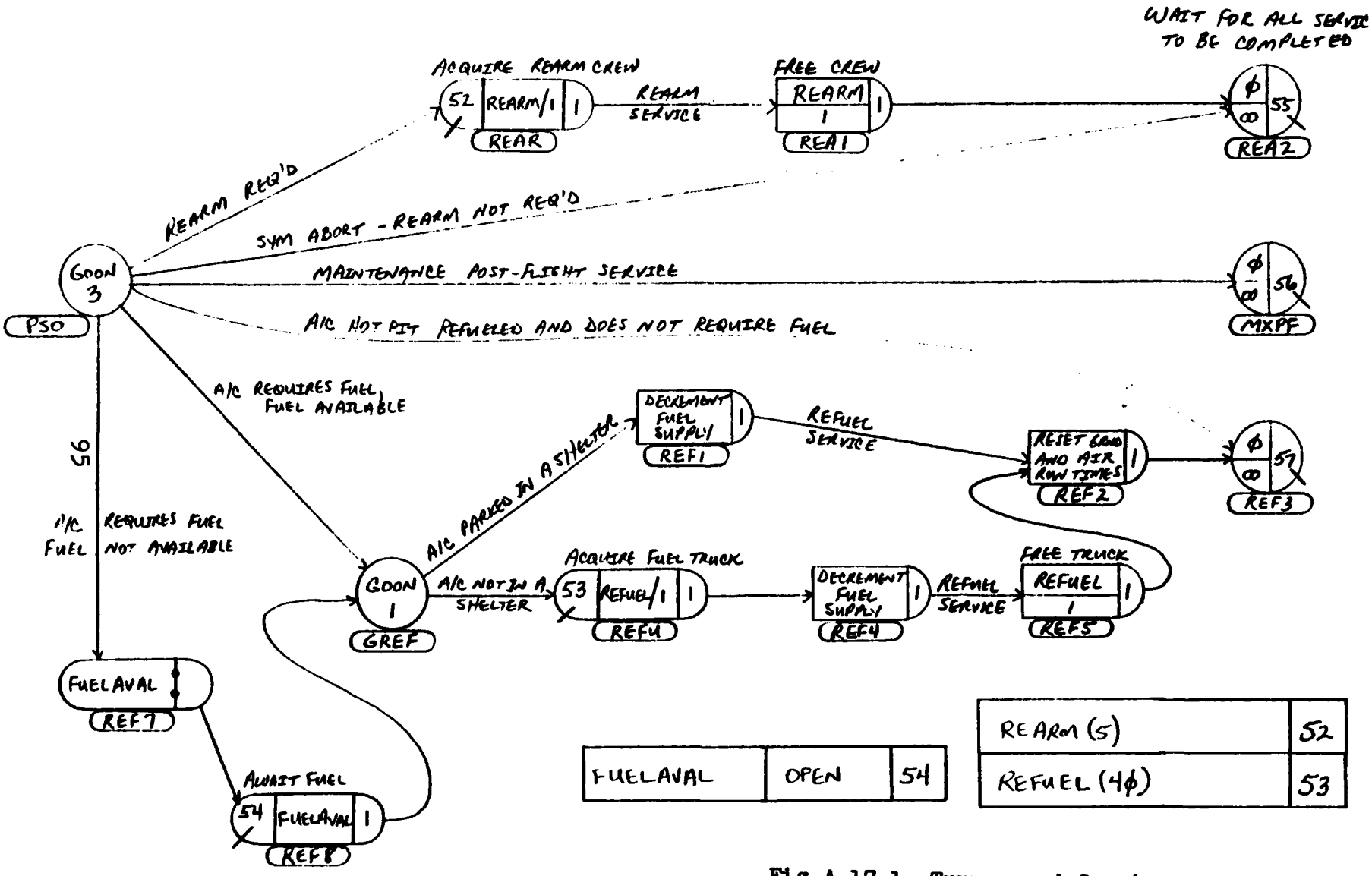


Fig A.17.1 Turnaround Service Operation

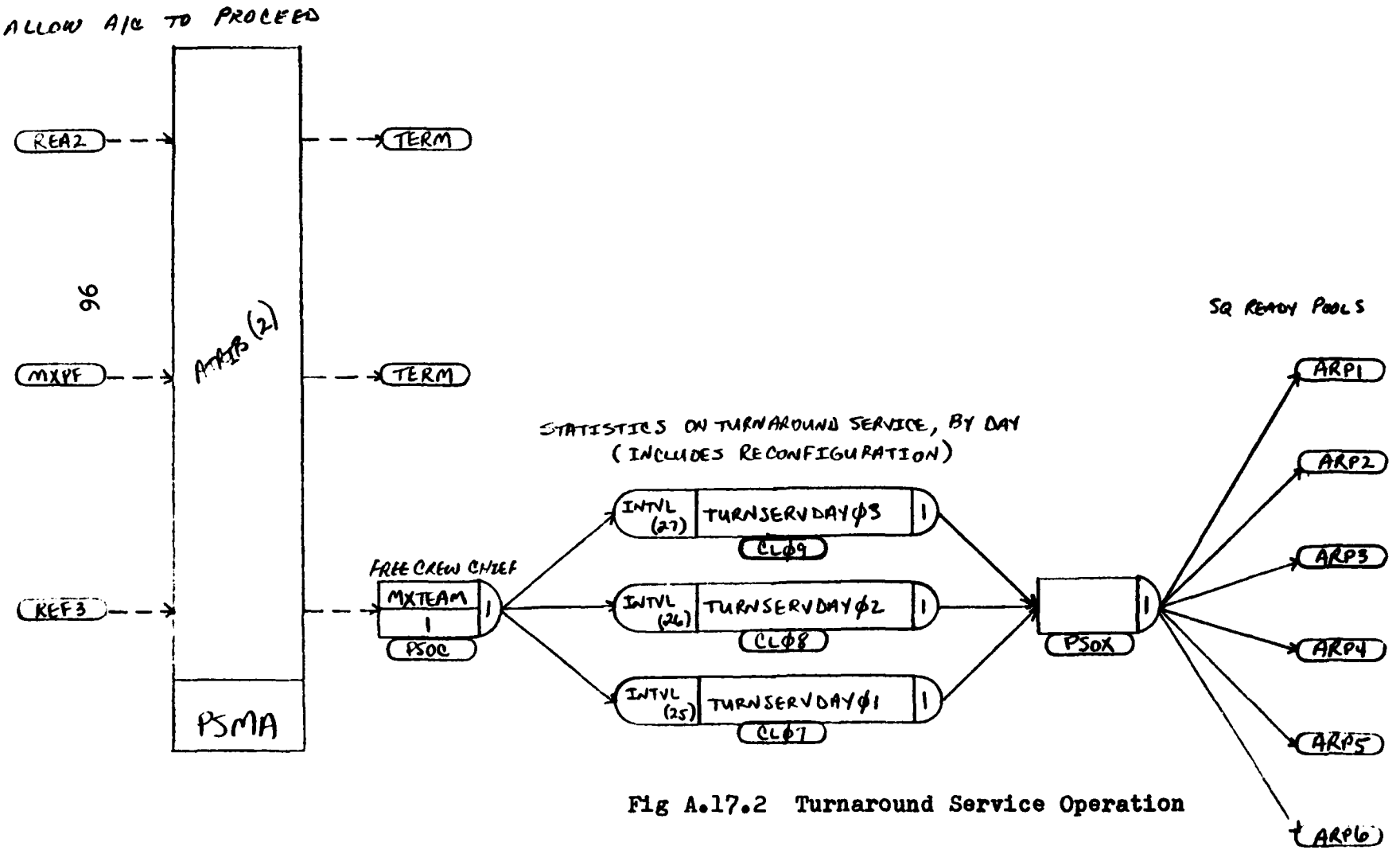


Fig A.17.2 Turnaround Service Operation

```
2497            ; MAINTENANCE CONTROL
2498           ;    A/C WHICH REQUIRE MAINTENANCE ARE INITIALLY DIVIDED INTO 3
2499           ; GROUPS- THOSE WITH FAILURE LEVELS LESS THAN 4 (MINOR), THOSE
2500           ; WITH FAILURE LEVELS EQUAL TO OR GREATER THAN 4 (MAJOR), AND
2501           ; THOSE WITH BATTLE DAMAGE.  A/C WITH MINOR PROBLEMS ARE SENT
2502           ; TO SQUADRON LEVEL MAINTENANCE FOR SERVICE.  A/C WITH BATTLE
2503           ; DAMAGE ARE EITHER SCRAPPED (USE FOR KBALL OR SEND TO DEPOT)
2504           ; OR THEY ARE GIVEN COMMENSURATE MAINTENANCE FAILURE CODES AND
2505           ; ROUTED ON TO MAINTENANCE CONTROL (MCON).  A/C WITH MAJOR
2506           ; PROBLEMS GO TO MCON.
2507           ;    AT MCON THE DECISION IS MADE ON WHERE TO REPAIR THE A/C WITH
2508           ; MAJOR MAINTENANCE PROBLEMS.  THE DECISION IS BASED ON THE
2509           ; FOLLOWING RULES-
2510           ;
2511           ;
2512           ; 1 - REPAIR AT WING IF A REQUIRED SHOP IS FREE.
2513           ;
2514           ; 2 - REPAIR WITH MMT IF A REQUIRED MMT IS FREE.
2515           ;
2516           ; 3 - WAIT FOR REPAIR AT WING IF WAITING SPACE IS FREE.
2517           ;
2518           ; 4 - GO TO SQUADRON MAINTENANCE AND REPAIR MINOR PROBLEMS,
2519           ;       THEN WAIT FOR MMT.
2520           ;
2521           ; AS IT IS THROUGHOUT THE MODEL, AN A/C CAN EITHER TAXI OR BE
2522           ; TOWED TO THE CORRECT FACILITY.
2523           ;
2524            MAIN ASSIGN,XX(95)=USERF(125),1;      MAINTENANCE CONTROL ROUTINE
2525                 ACT,,USERF(37).LT.4.AND.
2526                      ATRIB(16).EQ.0,SP;       TO SQ (NO BATTLE DAMAGE)
2527                 ACT,,ATRIB(16).EQ.0.AND.
2528                      USERF(37).GE.4,MCON;     TO MX CONTROL SORTING
2529                 ACT,TRIAG(10,15,30),
2530                     ATRIB(16).NE.0,BADA;      TO BATTLE DAMAGE ASSESSMENT
2531           ;
2532            SP   ASSIGN,ATRIB(3)=USERF(21),1;     GET A PARKING SPACE
2533                 ACT,USERF(65),
2534                     ATRIB(32).EQ.0,SPMX;      GET A CREW CHIEF
2535                 ACT,USERF(61),,SPT1;          GO TO ENGINE SHUT DOWN FOR SQ MX
2536           ;
2537            SPT1 ASSIGN,ATRIB(8)=TNOW-ATRIB(8),
2538                     ATRIB(7)=ATRIB(7)+ATRIB(8),
2539                     ATRIB(14)=ATRIB(14)+ATRIB(8),
2540                   XX(95)=USERF(124),
2541                     ATRIB(18)=USERF(51),2;    RESET CODES,ENGINE SHUT DOWN, SQ MX
2542                 ACT,,,PSEP;                   SEPARATE PILOT FROM A/C
2543                 ACT,,,SPMX;                   GET A CREW CHIEF
2544           ;
2545            SPMX AWAIT(98),MXTEAM/1,1;         WAIT FOR A CREW CHIEF
2546                 ACT,,,SMXC;
```

```
2547            ;
2548            SMXC GOON,1;                          BRANCH TO MMT OR LESSER REPAIRS
2549            ;                                       IF NO MMT IS AVAILABLE (IF REQ)
2550            ;
2551                    ACT,,USERF(112).EQ.1,DLMT;    GO TO MMT ROUTINE IF REQ'D
2552                    ACT,,,SMB;                    SQMX, MMT NOT REQ'D OR UNAVAIL
2553            ;
2554            DLMT GOON,1;
2555                    ACT/50,TRIAG(12,15,24),
2556                        ATRIB(10).EQ.1,MMT;       DOWNLOAD ORDNANCE
2557                    ACT,,,MMT;
2558            ;
2559            SMB  ASSIGN,ATRIB(17)=USERF(39),1;    ROUTE TO APPROPRIATE SQ MX AREA
2560                    ACT,,ATRIB(1).EQ.1,SM1;       A/C TO IT'S SQ MX AREA
2561                    ACT,,ATRIB(1).EQ.2,SM2;
2562                    ACT,,ATRIB(1).EQ.3,SM3;
2563                    ACT,,ATRIB(1).EQ.4,SM4;
2564                    ACT,,ATRIB(1).EQ.5,SM5;
2565                    ACT,,ATRIB(1).EQ.6,SM6;
2566            ;
2567            MCON GOON,1;                          ROUTE TO APPROPRIATE MX AREA
2568                    ACT,,USERF(111).EQ.1,MCW1;    TO WING MX IF AVAILABLE
2569            ;
2570                    ACT,,USERF(112).EQ.1,SP;      IF NO WING, TO MMT IF AVAILABLE
2571                    ACT,,NNQ(62).LT.2,MCW1;       NO WING OR MMT SO WING Q IF OPEN
2572                    ACT,,,SP;                     IF ALL ELSE FAILS, TO SQ FOR MMT
2573            ;
2574            MCW1 ASSIGN,ATRIB(3)=4,1;             PARKED AT WING MX
2575                    ACT,USERF(66),
2576                        ATRIB(32).EQ.0,DLWG;      ADD TOW TIME
2577                    ACT,USERF(62),,MCW2;          ADD TAXI TIME
2578            ;
2579            MCW2 ASSIGN,ATRIB(8)=TNOW-ATRIB(8),
2580                        ATRIB(7)=ATRIB(7)+ATRIB(8),
2581                        ATRIB(14)=ATRIB(14)+ATRIB(8),
2582                      XX(95)=USERF(124),
2583                        ATRIB(18)=USERF(51),2;    RESET CODES
2584                    ACT,,,DLWG;                   A/C INTO WING MX
2585                    ACT,,,PSEP;                   SEPARATE PILOT FROM A/C
2586            ;
2587            DLWG GOON,1;
2588                    ACT/49,TRIAG(12,15,24),
2589                          ATRIB(10).EQ.1,WG;      DOWNLOAD ORDNANCE
2590                    ACT,,,WG;                     PROCEED DIRECT TO WG MX
2591            ;
2592            BADA ASSIGN,ATRIB(18)=USERF(53),
2593                        XX(95)=USERF(124),2;      CHANGE MX FAIL CODE AS A FUNCTION
2594            ;                                       OF BATTLE DAMAGE
2595                    ACT,,USERF(37).GE.4.AND.
2596                         ATRIB(18).NE.999999,MCON;A/C IS REPAIRABLE
```

```
 2597            ;
2598                    ACT,,USERF(37).LT.4.AND.
2599                         ATRIB(18).NE.999999,SP;  TO SQ MX
2600                    ACT,,ATRIB(18).EQ.999999.AND.
2601                         ATRIB(32).NE.0.0,PSEP;   SEPARATE PILOT BEFORE JUNKING A/C
2602                    ACT,,ATRIB(18).EQ.999999,SRAP;TO FILE FOR DAILY CLEAN UP
2603            ;                                      (ATTRITED,CRASHED, OR NON-
2604            ;                                      REPAIRABLE)
2605            ;
2606             SRAP ASSIGN,ATRIB(32)=0,1;           REMOVE PILOT
2607                    ACT,,,CRSH;
2608            ;
```

WING MAINTENANCE

WGSHOP1 (2)	58
WGSHOP2 (2)	59
WGSHOP3 (2)	60
WGSHOP4 (2)	61

WGPOOL	OPEN	62

MMT MAINTENANCE

MMT1 (2)	63
MMT2 (2)	64
MMT3 (2)	65
MMT4 (2)	66
MMT5 (2)	67
MMT6 (2)	68

MMT POOL	OPEN	69

100

SQUADRON MAINTENANCE

SQ%MX1 (4)	WW
SQ%MX2 (4)	XX
SQ%MX3 (4)	YY
SQ%MX4 (4)	ZZ

% ≡ 1 THROUGH 6

WW ≡ (70, 74, 78, 82, 86, 90)

XX ≡ (71, 75, 79, 83, 87, 91)

YY ≡ (72, 76, 80, 84, 88, 92)

ZZ ≡ (73, 77, 81, 85, 89, 93)

Fig A.18.1 Maintenance Resources and Gates

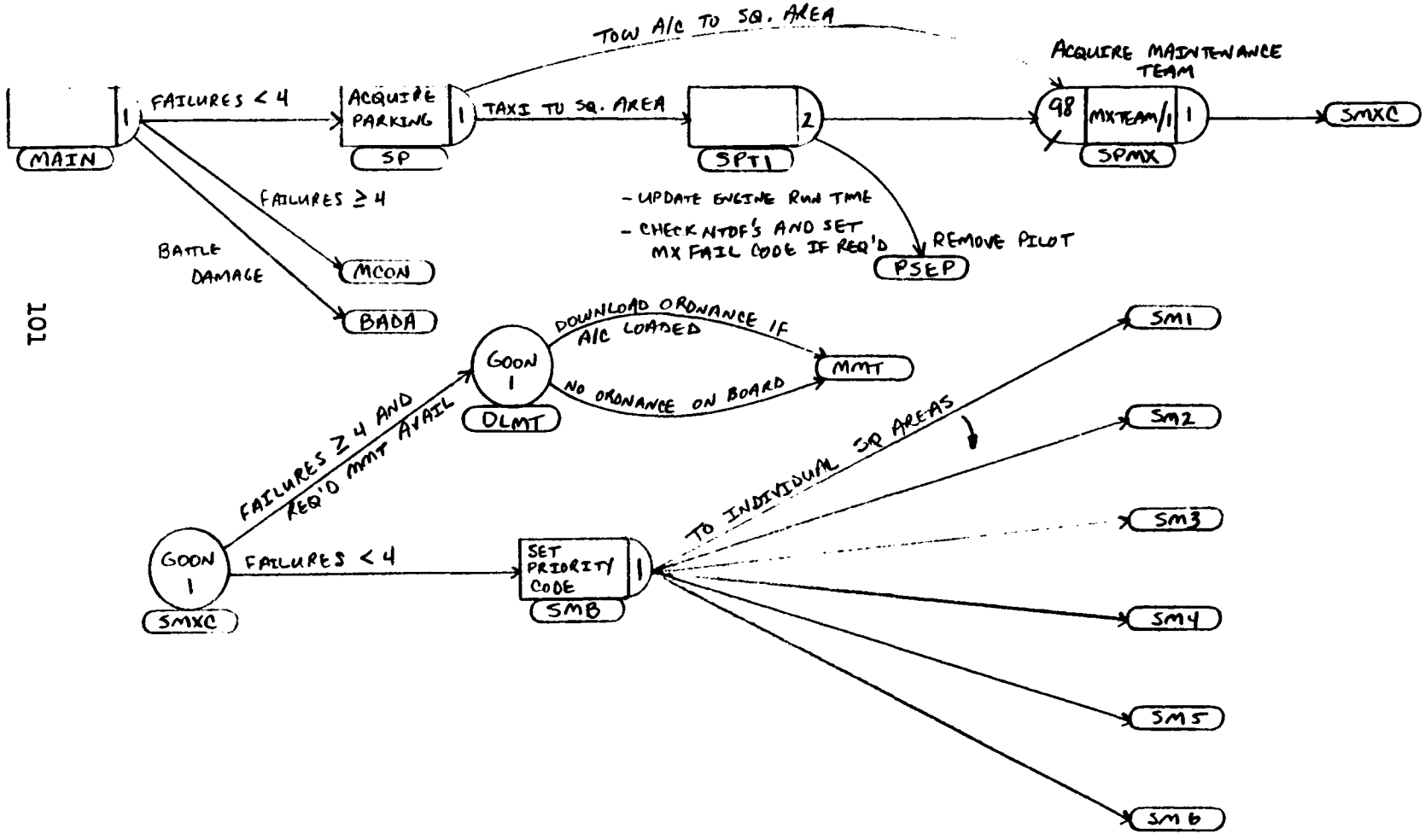


Fig A.18.2 Maintenance Control

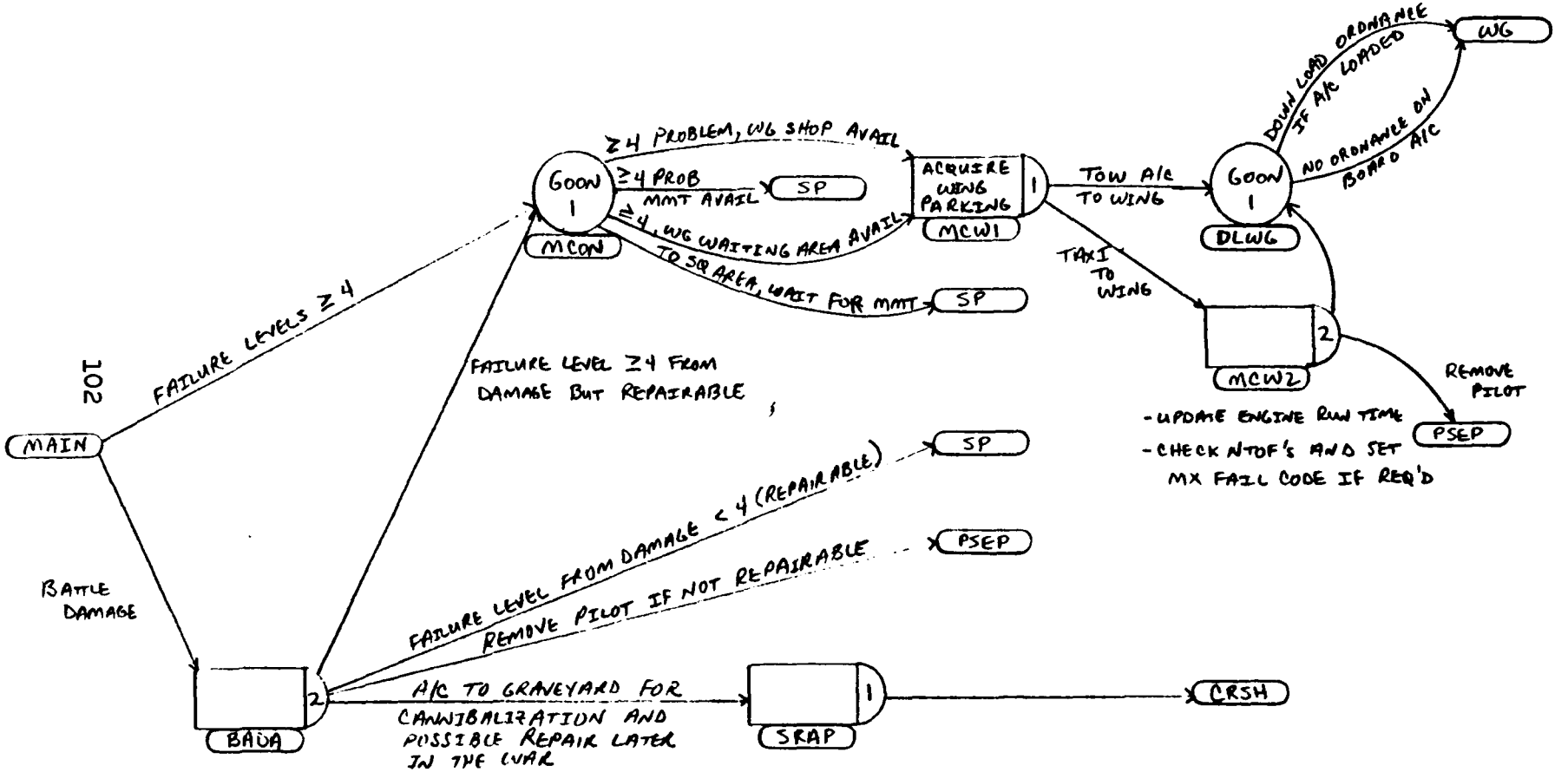


Fig A.18.3 Maintenance Control

```
; WING MAINTENANCE
;    AN A/C ARRIVING AT WING MAINTENANCE IS ASSIGNED A PRIORITY CODE
; BASED ON THE LEVEL AND NUMBER OF SYSTEMS REQUIRING REPAIR.  ALL
; LEVEL FOUR AND FIVE SYSTEMS ARE ADDED TO YIELD A NUMBER.  A/C
; ARE PROCESSED BASED ON LOW VALUE FIRST OF THE NUMBER. (FIX THE
; EASIEST ONES FIRST)  IF A SHOP IS OPEN THE A/C IS PROCESSED,
; ELSE IT GOES TO THE WAITING POOL.  WHEN AN A/C COMPLETES SERVICE
; IT FREES THE WING SHOP, RESETS ITS FAILURE CODE AND NTOF (FOR
; THE SYSTEMS REPAIRED) AND SIGNALS THE A/C WAITING IN THE WAITING
; POOL THAT A WING SHOP IS FREE.  IF A/C WAITING FOR A WING SHOP
; CAN NOT USE THE WING SHOP WHICH WAS FREED, THE A/C WAITING IN
; THE MMT POOL ARE NOTIFIED.  IF ANY OF THOSE A/C CAN USE THE OPEN
; WING SHOP, THE A/C IS TOWED TO WING.  ALL A/C UNABLE TO USE THE
; WING SHOP FROM BOTH WING AND MMT WAITING POOLS ARE RETURNED TO
; THE APPROPRIATE WAITING POOL (THE ONE THEY CAME FROM).
;    IF THE A/C WHICH FREED THE WING SHOP HAS BEEN COMPLETELY REPAIRED
; IT PROCEEDS TO TURNAROUND SERVICE.  IF IT STILL HAS A MAJOR PROB-
; LEM, IT TRYS TO GET INTO THE REQUIRED WING SHOP OR WAITS IN THE
; WING WAITING POOL.
;    IT IS ASSUMED THAT ALL MINOR PROBLEMS AN A/C MAY HAVE ARE REPAIRED
; WHILE THE A/C IS IN SERVICE AT WING.  NO ADDITIONAL DELAY IS
; ADDED FOR THIS SERVICE, THE FAILURE CODES AND NTOFS ARE JUST RESET.
;    THE WING SHOPS SERVICE THE FOLLOWING PROBLEMS-
;
;
; 1 - HYDRAULICS/PNEUMATICS AND AIRFRAME
;
; 2 - ENGINE/FUEL
;
; 3 - ELECTRICAL AND COMM/NAV/INSTRUMENTS/RADAR
;
; 4 - FIRE CONTROL/WEAPONS RELEASE
;
WG   ASSIGN,ATRIB(10)=0,
           ATRIB(17)=USERF(38),1;    WING MAINTENANCE ROUTINE
       ACT,,USERF(33).GE.4.AND.NNRSC(WGSHOP1).GT.0.OR.
           USERF(34).GE.4.AND.NNRSC(WGSHOP1).GT.0,
           WM11;                     HYDRAULICS OR AIRFRAME
       ACT,,USERF(32).GE.4.AND.NNRSC(WGSHOP2).GT.0,
           WM21;                     ENGINE/FUEL
       ACT,,USERF(31).GE.4.AND.NNRSC(WGSHOP3).GT.0.OR.
           USERF(35).GE.4.AND.NNRSC(WGSHOP3).GT.0,
           WM31;                     ELECTRICAL OR COMM/NAV/INST/RADAR
       ACT,,USERF(36).GE.4.AND.NNRSC(WGSHOP4).GT.0,
           WM41;                     FIRE CONTROL/WEAPONS RELEASE
       ACT,,,WMP1;                   GATE TO INSURE TRAP IN Q WMP2
;
WM11 AWAIT(58),WGSHOP1/1,1;          AIRFRAME
       ACT/7,USERF(81),,WM12;
;
```

```
2659            WM12 FREE,WCSHOP1/1,1;
2660                 ACT,,,WM13;
2661            ;
2662            WM13 ASSIGN,ATRIB(21)=ATRIB(7)+USERF(133),
2663                        ATRIB(18)=USERF(43),
2664                        ATRIB(22)=ATRIB(7)+USERF(134),
2665                        ATRIB(18)=USERF(44),1;
2666                 ACT,,USERF(37).GE.4,WMD1;
2667                 ACT,,USERF(32).GE.2,WM23;
2668                 ACT,,USERF(31).GE.2.OR.
2669                      USERF(35).GE.2,WM33;
2670                 ACT,,USERF(36).GE.2,WM43;
2671                 ACT,,,WMD1;
2672            ;
2673            WM21 AWAIT(59),WCSHOP2/1,1;          ENGINES/FUEL
2674                 ACT/8,USERF(82),,WM22;
2675            ;
2676            WM22 FREE,WCSHOP2/1,1;
2677                 ACT,,,WM23;
2678            ;
2679            WM23 ASSIGN,ATRIB(20)=ATRIB(7)+USERF(132),
2680                        ATRIB(18)=USERF(42),1;
2681                 ACT,,USERF(37).GE.4,WMD1;
2682                 ACT,,USERF(33).GE.2.OR.
2683                      USERF(34).GE.2,WM13;
2684                 ACT,,USERF(31).GE.2.OR.
2685                      USERF(35).GE.2,WM33;
2686                 ACT,,USERF(36).GE.2,WM43;
2687                 ACT,,,WMD1;
2688            ;
2689            WM31 AWAIT(60),WCSHOP3/1,1;          COMM/NAV/INST/RADIO/RADAR
2690                 ACT/9,USERF(83),,WM32;
2691            ;
2692            WM32 FREE,WCSHOP3/1,1;
2693                 ACT,,,WM33;
2694            ;
2695            WM33 ASSIGN,ATRIB(19)=ATRIB(7)+USERF(131),
2696                        ATRIB(18)=USERF(41),
2697                        ATRIB(23)=ATRIB(7)+USERF(135),
2698                        ATRIB(18)=USERF(45),1;
2699                 ACT,,USERF(37).GE.4,WMD1;
2700                 ACT,,USERF(33).GE.2.OR.
2701                      USERF(34).GE.2,WM13;
2702                 ACT,,USERF(32).GE.2,WM23;
2703                 ACT,,USERF(36).GE.2,WM43;
2704                 ACT,,,WMD1;
2705            ;
2706            WM41 AWAIT(61),WCSHOP4/1,1;          FIRE CONTROL/WEAPONS RELEASE
2707                 ACT/10,USERF(84),,WM42;
2708            ;
```

```
2709            WM42 FREE,WGSHOP4/1,1;
2710                 ACT,,,WM43;
2711            ;
2712             WM43 ASSIGN,ATRIB(24)=ATRIB(7)+USERF(136),
2713                        ATRIB(18)=USERF(46),1;
2714                 ACT,,USERF(37).GE.4,WMD1;
2715                 ACT,,USERF(33).GE.2.OR.
2716                      USERF(34).GE.2,WM13;
2717                 ACT,,USERF(32).GE.2,WM23;
2718                 ACT,,USERF(31).GE.2.OR.
2719                      USERF(35).GE.2,WM33;
2720                 ACT,,,WMD1;
2721            ;
2722             WMP1 CLOSE,WGPOOL,1;
2723                 ACT,,,WMP2;
2724            ;
2725             WMP2 AWAIT(62),WGPOOL,1;
2726                 ACT,,,WG;
2727            ;
2728             WMD1 OPEN,WGPOOL,1;
2729                 ACT,.0001,,WMD2;
2730            ;
2731             WMD2 OPEN,MMTPOOL,1;
2732                 ACT,.0001,USERF(37).GE.4,WG;
2733                 ACT,.0001,,WMD3;
2734             WMD3 ASSIGN,ATRIB(17)=0,ATRIB(18)=USERF(47),
2735                        ATRIB(3)=USERF(21),
2736                        ATRIB(13)=9,ATRIB(16)=0,1;
2737                 ACT,USERF(65),ATRIB(27).NE.0,CL12;
2738                 ACT,USERF(65),ATRIB(26).NE.0,CL11;
2739                 ACT,USERF(65),,CL10;
2740            ;                                    STATISTICS ON WING SERVICE
2741             CL12 COLCT,INTVL(27),WGSERVCOMPDAY03,,1;
2742                 ACT,,,MXTM;
2743             CL11 COLCT,INTVL(26),WGSERVCOMPDAY02,,1;
2744                 ACT,,,MXTM;
2745             CL10 COLCT,INTVL(25),WGSERVCOMPDAY01,,1;
2746                 ACT,,,MXTM;
2747            ;
```

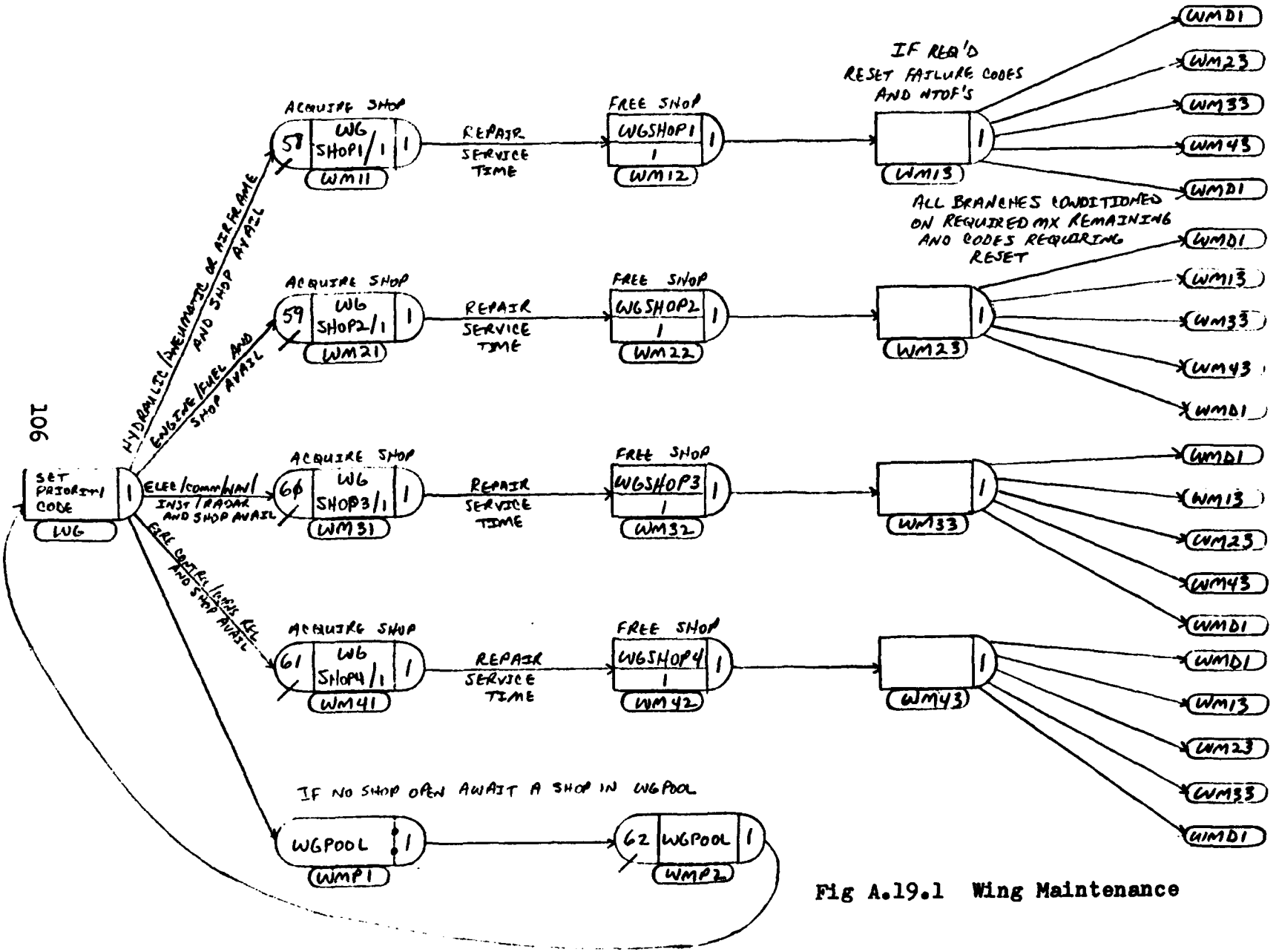


Fig A.19.1 Wing Maintenance

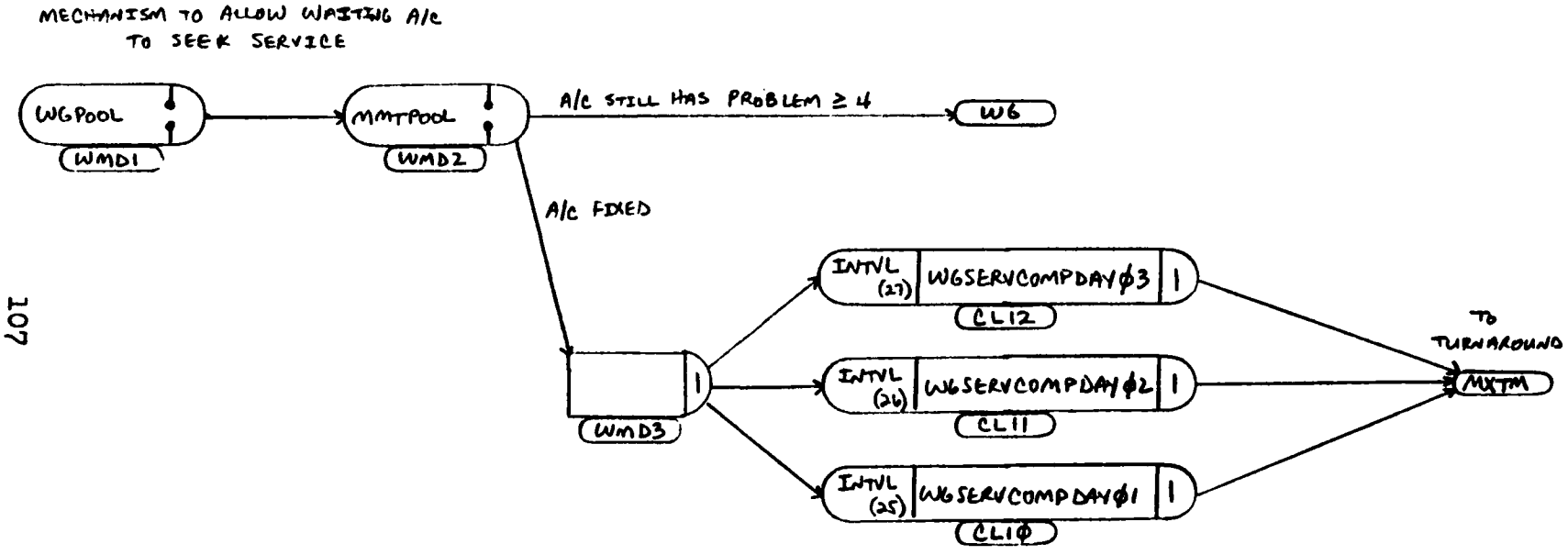


Fig A.19.2 Wing Maintenance

```
2748            ; MMT MAINTENANCE
2749           ;    AN A/C ARRIVING AT MMT MAINTENANCE IS ASSIGNED A PRIORITY PROCESS-
2750           ; ING CODE IN EXACTLY THE SAME WAY IT IS DONE AT WING (ADD ALL 4 AND
2751           ; 5 LEVEL FAILURES TOGETHER AND PROCESS ON LOW VALUE FIRST).  THE
2752           ; SYSTEM THEN REPAIRS THE QUICKEST FIRST.  IF A SHOP IS OPEN AND
2753           ; REQUIRED THE A/C IS PROCESSED IN THAT SHOP, ELSE IT IS SENT TO A
2754           ; WAITING POOL.  WHEN AN A/C COMPLETES SERVICE IT FREES THE MMT
2755           ; UNIT, RESETS THE FAILURE CODE AND NTOF FOR THE SYSTEM(S) REPAIRED,
2756           ; AND SIGNALS THE A/C WAITING IN THE MMT WAITING POOL THAT AN MMT
2757           ; UNIT IS FREE.  THE A/C IN THE POOL WITH THE HIGHEST PRIORITY
2758           ; (LOWEST VALUE) THAT REQUIRES THE MMT OBTAINS IT.  THE REST OF THE
2759           ; A/C ARE RETURNED TO THE MMT WAITING POOL.
2760           ;    OCCASIONLY, AN A/C WILL BE MOVED FROM THE MMT WAITING POOL TO THE
2761           ; WING MAINTENANCE FACILITY.  THIS HAPPENS WHEN A WING SHOP BECOMES
2762           ; FREE AND NO A/C WAITING AT WING REQUIRE THE SHOP, BUT AN A/C
2763           ; WAITING AT MMT DOES.
2764           ;    IF THE A/C FREEING THE MMT UNIT IS DONE, IT PROCEEDS TO TURNAROUND
2765           ; SERVICE.  IF THE A/C REQUIRES FURTHER SERVICE, IT TRYS TO OBTAIN
2766           ; THE DESIRED MMT UNIT WHICH IS APPROPRIATE FOR REMAINING MAJOR
2767           ; PROBLEMS.  IF ALL MAJOR REPAIRS ARE COMPLETED THE A/C IS SENT TO
2768           ; SQUADRON LEVEL MAINTENANCE IF IT REQUIRES ANY MINOR REPAIRS.
2769           ; UNLIKE WING SHOPS, MMTS DO NOT CONCURRENTLY REPAIR MINOR PROBLEMS.
2770           ;    MMT UNITS REPAIR THE FOLLOWING SYSTEMS-
2771           ;
2772           ;
2773           ; 1 - ELECTRICAL
2774           ;
2775           ; 2 - ENGINE/FUEL
2776           ;
2777           ; 3 - HYDRAULICS/PNEUMATICS
2778           ;
2779           ; 4 - AIRFRAME
2780           ;
2781           ; 5 - COMM/NAV/INSTRUMENTS/RADAR
2782           ;
2783           ; 6 - FIRE CONTROL/WEAPONS RELEASE
2784           ;
2785            MMT  ASSIGN,ATRIB(10)=0,
2786                        ATRIB(17)=USERF(38),1;    MOBILE MAINTENANCE TEAM ROUTINE
2787                 ACT,,USERF(31).GE.4.AND.
2788                      NNRSC(MMT1).GT.0,MM11;   IF REQ'D AND AVAIL, GO TO
2789                 ACT,,USERF(32).GE.4.AND.
2790                      NNRSC(MMT2).GT.0,MM21;
2791                 ACT,,USERF(33).GE.4.AND.
2792                      NNRSC(MMT3).GT.0,MM31;
2793                 ACT,,USERF(34).GE.4.AND.
2794                      NNRSC(MMT4).GT.0,MM41;
2795                 ACT,,USERF(35).GE.4.AND.
2796                      NNRSC(MMT5).GT.0,MM51;
2797                 ACT,,USERF(36).GE.4.AND.
```

```
                    NNRSC(MMT6).GT.0,MM61;
              ACT,,,MMP1;
            ;
            MM11 AWAIT(63),MMT1/1,1;              ELECTRICAL
              ACT/11,USERF(91),,MM12;
            ;
            MM12 FREE,MMT1/1,1;
              ACT,,,MM13;
            ;
            ;
            MM13 ASSIGN,ATRIB(18)=USERF(41),
                  ATRIB(19)=ATRIB(7)+USERF(131),1;RESET FAIL CODE 0,NTOF RESET
              ACT,,,MOPL;                         TURNAROUND SERVICE PREP
            ;
            MM21 AWAIT(64),MMT2/1,1;
              ACT/12,USERF(92),,MM22;             ENGINE/FUEL
            ;
            MM22 FREE,MMT2/1,1;
              ACT,,,MM23;
            ;
            ;
            MM23 ASSIGN,ATRIB(18)=USERF(42),
                  ATRIB(20)=ATRIB(7)+USERF(132),1;RESET FAIL CODE 0,NTOF RESET
              ACT,,,MOPL;                         TURNAROUND SERVICE PREP
            ;
            MM31 AWAIT(65),MMT3/1,1;              HYDRAULICS/PNEUMATICS
              ACT/13,USERF(93),,MM32;
            ;
            MM32 FREE,MMT3/1,1;
              ACT,,,MM33;
            ;
            ;
            MM33 ASSIGN,ATRIB(18)=USERF(43),
                  ATRIB(21)=ATRIB(7)+USERF(133),1;RESET FAIL CODE 0,NTOF RESET
              ACT,,,MOPL;                         TURNAROUND SERVICE PREP
            ;
            MM41 AWAIT(66),MMT4/1,1;              AIRFRAME(STRUTS,TIRES)
              ACT/14,USERF(94),,MM42;
            ;
            MM42 FREE,MMT4/1,1;
              ACT,,,MM43;
            ;
            ;
            MM43 ASSIGN,ATRIB(18)=USERF(44),
                  ATRIB(22)=ATRIB(7)+USERF(134),1;RESET FAIL CODE 0,NTOF RESET
              ACT,,,MOPL;                         TURNAROUND SERVICE PREP
            ;
            MM51 AWAIT(67),MMT5/1,1;              COMM/NAV/INST/RADIO/RADAR
              ACT/15,USERF(95),,MM52;
            ;
```

```
2848            MM52 FREE,MMT5/1,1;
2849                 ACT,,,MM53;
2850            ;
2851            ;
2852            MM53 ASSIGN,ATRIB(18)=USERF(45),
2853                  ATRIB(23)=ATRIB(7)+USERF(135),1;RESET FAIL CODE Ø,NTOF RESET
2854                 ACT,,,MOPL;                    TURNAROUND SERVICE PREP
2855            ;
2856            MM61 AWAIT(68),MMT6/1,1;
2857                 ACT/16,USERF(96),,MM62;        FIRE CONTROL/WEAPONS RELEASE
2858            ;
2859            MM62 FREE,MMT6/1,1;
2860                 ACT,,,MM63;
2861            ;
2862            ;
2863            MM63 ASSIGN,ATRIB(18)=USERF(46),
2864                  ATRIB(24)=ATRIB(7)+USERF(136),1;RESET FAIL CODE Ø,NTOF RESET
2865                 ACT,,,MOPL;                    TURNAROUND SERVICE PREP
2866            ;
2867            MMP1 CLOSE,MMTPOOL,1;
2868                 ACT,,,MMP2;
2869            ;
2870            MMP2 AWAIT(69),MMTPOOL,1;           WAIT FOR MMT SERVICE
2871                 ACT,.ØØØ2,,MMPG;                 A/C HAVE A CHANCE TO GO FOR A
2872            ;                                   MAJOR REPAIR IF SPACE AVAILABLE
2873            ;                                   IN A WING SHOP THEY REQUIRE
2874            ;                                   ELSE THEY LOOP BACK TO THE MMT
2875            ;                                   ENTRY ROUTINE
2876            MMPG GOON,1;
2877                 ACT,,USERF(113).EQ.1,MMP3;
2878                 ACT,,,MMT;
2879            ;
2880            MMP3 ASSIGN,ATRIB(3)=USERF(22),
2881                  ATRIB(3)=4,XX(93)=1,1;        UNPARK AT SQ,SET WING PARK,CLOSE
2882            ;                                     PATHWAY TO WING
2883                 ACT,.ØØØ1,,MMP4;               DELAY WHILE OTHER A/C IN THE
2884            ;                                     MMTPOOL LOOP BACK TO MMT ENTRY
2885            MMP4 ASSIGN,ATRIB(13)=ATRIB(1),
2886                    XX(93)=Ø,1;                 RESET PATH TO WC OPEN
2887                 ACT,USERF(66),,WC;
2888            ;
2889            MOPL OPEN,MMTPOOL,1;
2890                 ACT,.ØØØ1,USERF(37).GE.2,SMXC;
2891                 ACT,.ØØØ1,,MMRS;
2892            ;
2893            MMRS ASSIGN,ATRIB(16)=Ø,
2894                  ATRIB(17)=Ø,1;                RESET BATTLE DAMAGE AND SYM ABORT CODE
2895                 ACT,,ATRIB(27).NE.Ø,CL15;
2896                 ACT,,ATRIB(26).NE.Ø,CL14;
2897                 ACT,,,CL13;
```

```
 2898            ;                                        STATISTICS ON MMT SERVICE
2899             CL15 COLCT,INTVL(27),SQSERVCOMPDAY03,,1;
2900                    ACT,,,MXT1;
2901             ;
2902             CL14 COLCT,INTVL(26),SQSERVCOMPDAY02,,1;
2903                    ACT,,,MXT1;
2904             ;
2905             CL13 COLCT,INTVL(25),SQSERVCOMPDAY01,,1;
2906                    ACT,,,MXT1;
2907             ;
```

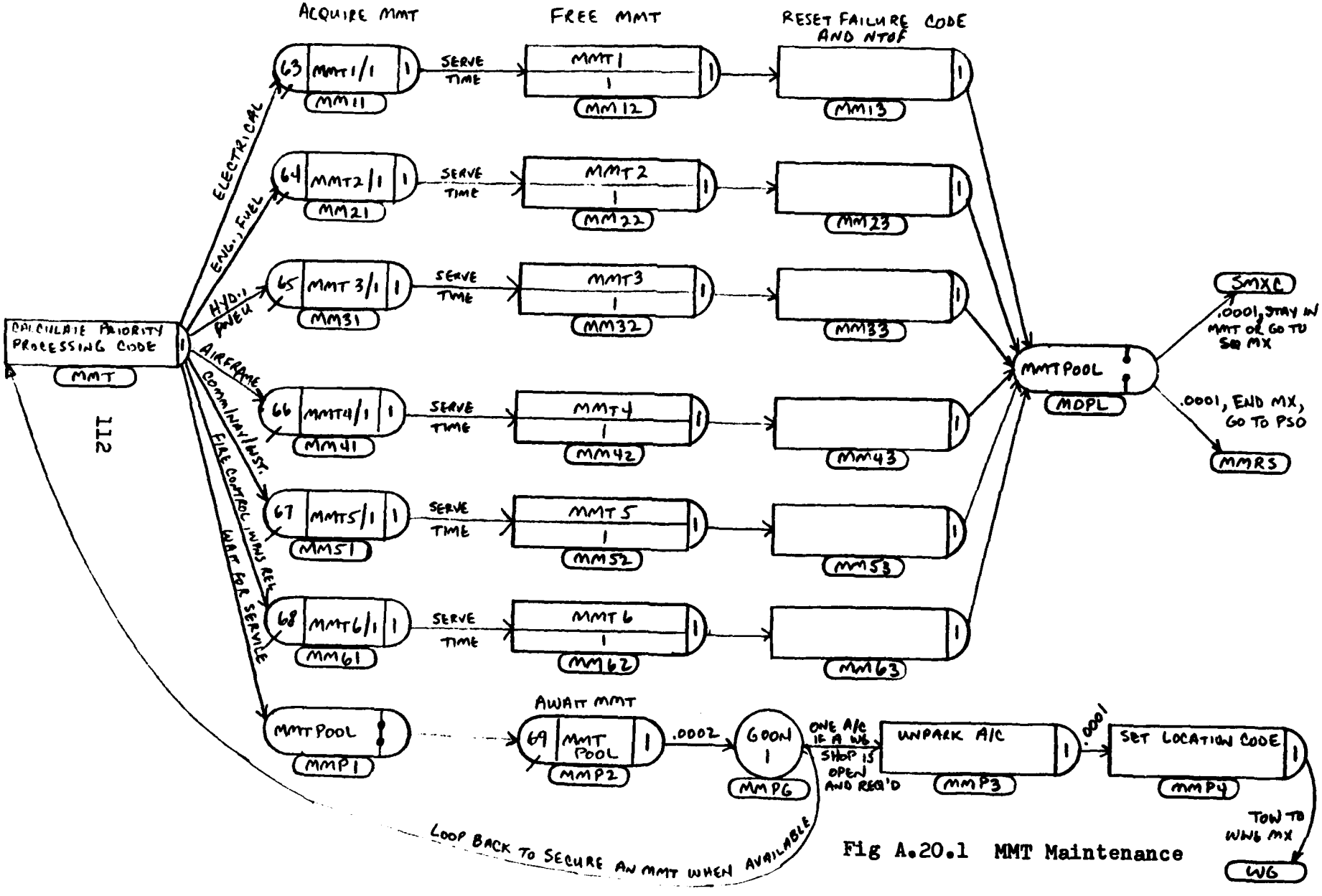


Fig A.20.1 MMT Maintenance

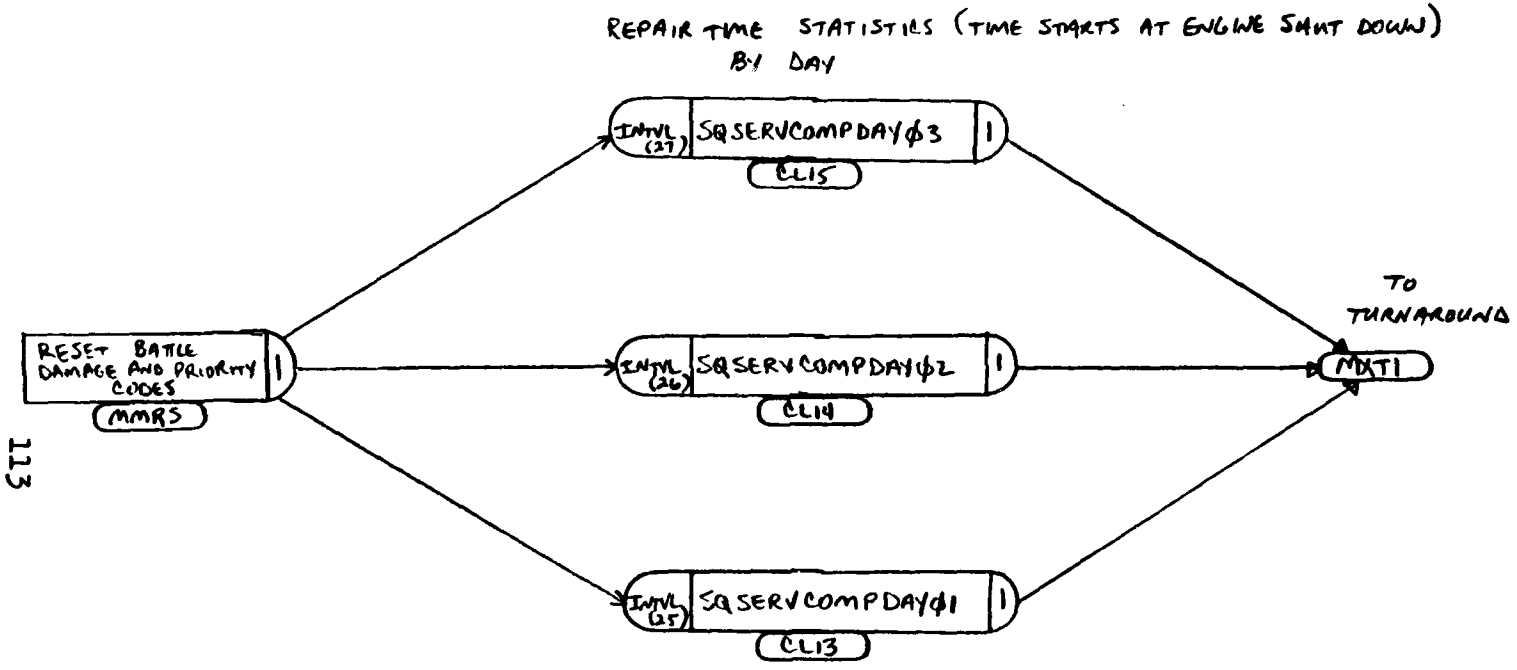


Fig A.20.2 MMT Maintenance

```
2908          ; SQUADRON MAINTENANCE
2909          ;    A/C WITH MINOR PROBLEMS ARE REPAIRED CONCURRENTLY.  THIS DIFFERS
2910          ; FROM WING AND MMT.  THE A/C PROCESS THROUGH SQUADRON MAINTENANCE
2911          ; RECEIVING THE SERVICE REQUIRED.  WHEN ALL SERVICE IS COMPLETED,
2912          ; THE FAILURE CODE AND NTOF (IF FAILURE WAS LEVEL 2 OR 3) IS RESET.
2913          ; A/C THAT STILL HAVE A MAJOR PROBLEM ARE SENT TO OBTAIN AN MMT
2914          ; UNIT.  THOSE A/C THAT ARE FINISHED WITH MAINTENANCE ARE SENT TO
2915          ; TURNAROUND SERVOCE.  EACH SQUADRON SHOP HAS ITS OWN SPECIALTY-
2916          ;
2917          ; 1 - HYDRAULICS/PNEUMATICS AND AIRFRAME
2918          ;
2919          ; 2 - ENGINE/FUEL
2920          ;
2921          ; 3 - ELECTRICAL AND COMM/NAV/INSTRUMENTS/RADAR
2922          ;
2923          ; 4 - FIRE CONTROL/WEAPONS RELEASE
2924          ;
2925           SM1  GOON,4;                          SQUADRON MAINTENANCE ROUTINE
2926                  ACT,,USERF(33).EQ.2.OR.
2927                   USERF(33).EQ.3.OR.
2928                   USERF(34).EQ.2.OR.
2929                   USERF(34).EQ.3,SQ11;         HYDRAULICS/PNEUMATICS OR AIRFRAME
2930                  ACT,,USERF(32).EQ.2.OR.
2931                   USERF(32).EQ.3,SQ12;         ENGINE/FUEL
2932                  ACT,,USERF(31).EQ.2.OR.
2933                   USERF(31).EQ.3.OR.
2934                   USERF(35).EQ.2.OR.
2935                   USERF(35).EQ.3,SQ13;         ELECTRICAL OR COMM/NAV/INST/RADAR
2936                  ACT,,USERF(36).EQ.2.OR.
2937                   USERF(36).EQ.3,SQ14;         FIRE CONTROL/WEAPONS RELEASE
2938                  ACT,,USERF(33).NE.2.AND.USERF(33).NE.3.AND.
2939                       USERF(34).NE.2.AND.USERF(34).NE.3,SQQ1;
2940                  ACT,,USERF(32).NE.2.AND.USERF(32).NE.3,SQQ2;
2941                  ACT,,USERF(31).NE.2.AND.USERF(31).NE.3.AND.
2942                       USERF(35).NE.2.AND.USERF(35).NE.3,SQQ3;
2943                  ACT,,USERF(36).NE.2.AND.USERF(36).NE.3,SQQ4;
2944          ;
2945           SQ11 AWAIT(70),SQ1MX1/1,1;            WAIT FOR SERVICE
2946                  ACT/17,USERF(101),,SQ15;
2947          ;
2948           SQ15 FREE,SQ1MX1/1,1;
2949                  ACT,,,SQQ1;
2950          ;
2951           SQ12 AWAIT(71),SQ1MX2/1,1;
2952                  ACT/18,USERF(102),,SQ16;
2953          ;
2954           SQ16 FREE,SQ1MX2/1,1;
2955                  ACT,,,SQQ2;
2956          ;
2957           SQ13 AWAIT(72),SQ1MX3/1,1;
```

```
2958                    ACT/19,USERF(103),,SQ17;
2959            ;
2960             SQ17 FREE,SQ1MX3/1,1;
2961                    ACT,,,SQQ3;
2962            ;
2963             SQ14 AWAIT(73),SQ1MX4/1,1;
2964                    ACT/20,USERF(104),,SQ18;
2965            ;
2966             SQ18 FREE,SQ1MX4/1,1;
2967                    ACT,,,SQQ4;
2968             SM2  GOON,4;                          SQUADRON MAINTENANCE ROUTINE
2969                    ACT,,USERF(33).EQ.2.OR.
2970                     USERF(33).EQ.3.OR.
2971                     USERF(34).EQ.2.OR.
2972                     USERF(34).EQ.3,SQ21;          HYDRAULICS/PNEUMATICS OR AIRFRAME
2973                    ACT,,USERF(32).EQ.2.OR.
2974                     USERF(32).EQ.3,SQ22;          ENGINE/FUEL
2975                    ACT,,USERF(31).EQ.2.OR.
2976                     USERF(31).EQ.3.OR.
2977                     USERF(35).EQ.2.OR.
2978                     USERF(35).EQ.3,SQ23;          ELECTRICAL OR COMM/NAV/INST/RADAR
2979                    ACT,,USERF(36).EQ.2.OR.
2980                     USERF(36).EQ.3,SQ24;          FIRE CONTROL/WEAPONS RELEASE
2981                    ACT,,USERF(33).NE.2.AND.USERF(33).NE.3.AND.
2982                         USERF(34).NE.2.AND.USERF(34).NE.3,SQQ1;
2983                    ACT,,USERF(32).NE.2.AND.USERF(32).NE.3,SQQ2;
2984                    ACT,,USERF(31).NE.2.AND.USERF(31).NE.3.AND.
2985                         USERF(35).NE.2.AND.USERF(35).NE.3,SQQ3;
2986                    ACT,,USERF(36).NE.2.AND.USERF(36).NE.3,SQQ4;
2987            ;
2988             SQ21 AWAIT(74),SQ2MX1/1,1;            WAIT FOR SERVICE
2989                    ACT/21,USERF(101),,SQ25;
2990            ;
2991             SQ25 FREE,SQ2MX1/1,1;
2992                    ACT,,,SQQ1;
2993            ;
2994             SQ22 AWAIT(75),SQ2MX2/1,1;
2995                    ACT/22,USERF(102),,SQ26;
2996            ;
2997             SQ26 FREE,SQ2MX2/1,1;
2998                    ACT,,,SQQ2;
2999            ;
3000             SQ23 AWAIT(76),SQ2MX3/1,1;
3001                    ACT/23,USERF(103),,SQ27;
3002            ;
3003             SQ27 FREE,SQ2MX3/1,1;
3004                    ACT,,,SQQ3;
3005            ;
3006             SQ24 AWAIT(77),SQ2MX4/1,1;
3007                    ACT/24,USERF(104),,SQ28;
```

```
;
SQ28 FREE,SQ2MX4/1,1;
       ACT,,SQQ4;
SM3  GOON,4;                              SQUADRON MAINTENANCE ROUTINE
       ACT,,USERF(33).EQ.2.OR.
        USERF(33).EQ.3.OR.
        USERF(34).EQ.2.OR.
        USERF(34).EQ.3,SQ31;              HYDRAULICS/PNEUMATICS OR AIRFRAME
       ACT,,USERF(32).EQ.2.OR.
        USERF(32).EQ.3,SQ32;              ENGINE/FUEL
       ACT,,USERF(31).EQ.2.OR.
        USERF(31).EQ.3.OR.
        USERF(35).EQ.2.OR.
        USERF(35).EQ.3,SQ33;              ELECTRICAL OR COMM/NAV/INST/RADAR
       ACT,,USERF(36).EQ.2.OR.
        USERF(36).EQ.3,SQ34;              FIRE CONTROL/WEAPONS RELEASE
       ACT,,USERF(33).NE.2.AND.USERF(33).NE.3.AND.
            USERF(34).NE.2.AND.USERF(34).NE.3,SQQ1;
       ACT,,USERF(32).NE.2.AND.USERF(32).NE.3,SQQ2;
       ACT,,USERF(31).NE.2.AND.USERF(31).NE.3.AND.
            USERF(35).NE.2.AND.USERF(35).NE.3,SQQ3;
       ACT,,USERF(36).NE.2.AND.USERF(36).NE.3,SQQ4;
;
SQ31 AWAIT(78),SQ3MX1/1,1;                WAIT FOR SERVICE
       ACT/25,USERF(101),,SQ35;
;
SQ35 FREE,SQ3MX1/1,1;
       ACT,,,SQQ1;
;
SQ32 AWAIT(79),SQ3MX2/1,1;
       ACT/26,USERF(102),,SQ36;
;
SQ36 FREE,SQ3MX2/1,1;
       ACT,,,SQQ2;
;
SQ33 AWAIT(80),SQ3MX3/1,1;
       ACT/27,USERF(103),,SQ37;
;
SQ37 FREE,SQ3MX3/1,1;
       ACT,,,SQQ3;
;
SQ34 AWAIT(81),SQ3MX4/1,1;
       ACT/28,USERF(104),,SQ38;
;
SQ38 FREE,SQ3MX4/1,1;
       ACT,,,SQQ4;
SM4  GOON,4;                              SQUADRON MAINTENANCE ROUTINE
       ACT,,USERF(33).EQ.2.OR.
        USERF(33).EQ.3.OR.
          USERF(34).EQ.2.OR.
```

```
3058                    USERF(34).EQ.3,SQ41;        HYDRAULICS/PNEUMATICS OR AIRFRAME
3059                  ACT,,USERF(32).EQ.2.OR.
3060                   USERF(32).EQ.3,SQ42;        ENGINE/FUEL
3061                  ACT,,USERF(31).EQ.2.OR.
3062                   USERF(31).EQ.3.OR.
3063                   USERF(35).EQ.2.OR.
3064                   USERF(35).EQ.3,SQ43;        ELECTRICAL OR COMM/NAV/INST/RADAR
3065                  ACT,,USERF(36).EQ.2.OR.
3066                   USERF(36).EQ.3,SQ44;        FIRE CONTROL/WEAPONS RELEASE
3067                  ACT,,USERF(33).NE.2.AND.USERF(33).NE.3.AND.
3068                       USERF(34).NE.2.AND.USERF(34).NE.3,SQQ1;
3069                  ACT,,USERF(32).NE.2.AND.USERF(32).NE.3,SQQ2;
3070                  ACT,,USERF(31).NE.2.AND.USERF(31).NE.3.AND.
3071                       USERF(35).NE.2.AND.USERF(35).NE.3,SQQ3;
3072                  ACT,,USERF(36).NE.2.AND.USERF(36).NE.3,SQQ4;
3073              ;
3074               SQ41 AWAIT(82),SQ4MX1/1,1;          WAIT FOR SERVICE
3075                      ACT/29,USERF(101),,SQ45;
3076              ;
3077               SQ45 FREE,SQ4MX1/1,1;
3078                      ACT,,,SQQ1;
3079              ;
3080               SQ42 AWAIT(83),SQ4MX2/1,1;
3081                      ACT/30,USERF(102),,SQ46;
3082              ;
3083               SQ46 FREE,SQ4MX2/1,1;
3084                      ACT,,,SQQ2;
3085              ;
3086               SQ43 AWAIT(84),SQ4MX3/1,1;
3087                      ACT/31,USERF(103),,SQ47;
3088              ;
3089               SQ47 FREE,SQ4MX3/1,1;
3090                      ACT,,,SQQ3;
3091              ;
3092               SQ44 AWAIT(85),SQ4MX4/1,1;
3093                      ACT/32,USERF(104),,SQ48;
3094              ;
3095               SQ48 FREE,SQ4MX4/1,1;
3096                      ACT,,,SQQ4;
3097               SM5  GOON,4;                        SQUADRON MAINTENANCE ROUTINE
3098                      ACT,,USERF(33).EQ.2.OR.
3099                       USERF(33).EQ.3.OR.
3100                       USERF(34).EQ.2.OR.
3101                       USERF(34).EQ.3,SQ51;        HYDRAULICS/PNEUMATICS OR AIRFRAME
3102                      ACT,,USERF(32).EQ.2.OR.
3103                       USERF(32).EQ.3,SQ52;        ENGINE/FUEL
3104                      ACT,,USERF(31).EQ.2.OR.
3105                       USERF(31).EQ.3.OR.
3106                       USERF(35).EQ.2.OR.
3107                       USERF(35).EQ.3,SQ53;        ELECTRICAL OR COMM/NAV/INST/RADAR
```

```
3108                    ACT,,USERF(36).EQ.2.OR.
3109                    USERF(36).EQ.3,SQ54;        FIRE CONTROL/WEAPONS RELEASE
3110                   ACT,,USERF(33).NE.2.AND.USERF(33).NE.3.AND.
3111                        USERF(34).NE.2.AND.USERF(34).NE.3,SQQ1;
3112                   ACT,,USERF(32).NE.2.AND.USERF(32).NE.3,SQQ2;
3113                   ACT,,USERF(31).NE.2.AND.USERF(31).NE.3.AND.
3114                        USERF(35).NE.2.AND.USERF(35).NE.3,SQQ3;
3115                   ACT,,USERF(36).NE.2.AND.USERF(36).NE.3,SQQ4;
3116            ;
3117            SQ51 AWAIT(86),SQ5MX1/1,1;          WAIT FOR SERVICE
3118                   ACT/33,USERF(101),,SQ55;
3119            ;
3120            SQ55 FREE,SQ5MX1/1,1;
3121                   ACT,,,SQQ1;
3122            ;
3123            SQ52 AWAIT(87),SQ5MX2/1,1;
3124                   ACT/34,USERF(102),,SQ56;
3125            ;
3126            SQ56 FREE,SQ5MX2/1,1;
3127                   ACT,,,SQQ2;
3128            ;
3129            SQ53 AWAIT(88),SQ5MX3/1,1;
3130                   ACT/35,USERF(103),,SQ57;
3131            ;
3132            SQ57 FREE,SQ5MX3/1,1;
3133                   ACT,,,SQQ3;
3134            ;
3135            SQ54 AWAIT(89),SQ5MX4/1,1;
3136                   ACT/36,USERF(104),,SQ58;
3137            ;
3138            SQ58 FREE,SQ5MX4/1,1;
3139                   ACT,,,SQQ4;
3140            SM6  GOON,4;                        SQUADRON MAINTENANCE ROUTINE
3141                   ACT,,USERF(33).EQ.2.OR.
3142                    USERF(33).EQ.3.OR.
3143                    USERF(34).EQ.2.OR.
3144                    USERF(34).EQ.3,SQ61;        HYDRAULICS/PNEUMATICS OR AIRFRAME
3145                   ACT,,USERF(32).EQ.2.OR.
3146                    USERF(32).EQ.3,SQ62;        ENGINE/FUEL
3147                   ACT,,USERF(31).EQ.2.OR.
3148                    USERF(31).EQ.3.OR.
3149                    USERF(35).EQ.2.OR.
3150                    USERF(35).EQ.3,SQ63;        ELECTRICAL OR COMM/NAV/INST/RADAR
3151                   ACT,,USERF(36).EQ.2.OR.
3152                    USERF(36).EQ.3,SQ64;        FIRE CONTROL/WEAPONS RELEASE
3153                   ACT,,USERF(33).NE.2.AND.USERF(33).NE.3.AND.
3154                        USERF(34).NE.2.AND.USERF(34).NE.3,SQQ1;
3155                   ACT,,USERF(32).NE.2.AND.USERF(32).NE.3,SQQ2;
3156                   ACT,,USERF(31).NE.2.AND.USERF(31).NE.3.AND.
3157                        USERF(35).NE.2.AND.USERF(35).NE.3,SQQ3;
```

```
3158                ACT,,USERF(36).NE.2.AND.USERF(36).NE.3,SQQ4;
3159            ;
3160             SQ61 AWAIT(90),SQ6MX1/1,1;          WAIT FOR SERVICE
3161                    ACT/37,USERF(101),,SQ65;
3162            ;
3163             SQ65 FREE,SQ6MX1/1,1;
3164                    ACT,,,SQQ1;
3165            ;
3166             SQ62 AWAIT(91),SQ6MX2/1,1;
3167                    ACT/38,USERF(102),,SQ66;
3168            ;
3169             SQ66 FREE,SQ6MX2/1,1;
3170                    ACT,,,SQQ2;
3171            ;
3172             SQ63 AWAIT(92),SQ6MX3/1,1;
3173                    ACT/39,USERF(103),,SQ67;
3174            ;
3175             SQ67 FREE,SQ6MX3/1,1;
3176                    ACT,,,SQQ3;
3177            ;
3178             SQ64 AWAIT(93),SQ6MX4/1,1;
3179                    ACT/40,USERF(104),,SQ68;
3180            ;
3181             SQ68 FREE,SQ6MX4/1,1;
3182                    ACT,,,SQQ4;
3183             SQQ1 QUEUE(94),,,,SQMA;             Q'S BEFORE MATCH IN COMMON
3184            ;
3185             SQQ2 QUEUE(95),,,,SQMA;               PROCESSOR TO PUT AN A/C BACK
3186            ;
3187             SQQ3 QUEUE(96),,,,SQMA;               TOGETHER AFTER IT WAS SPLIT UP
3188            ;
3189             SQQ4 QUEUE(97),,,,SQMA;               AND RUN THRU SQ MAINTENANCE
3190            ;
3191             SQMA MATCH,2,SQQ1/REDO,SQQ2/TERM,
3192                     SQQ3/TERM,SQQ4/TERM;
3193            ;
3194             TERM GOON,1;                        THE THREE REDUNDANT A/C ENTITIES
3195                  TERMINATE;                       ARE DESTROYED AND THE ENTITY
3196            ;                                      CONTINUES AS A SINGLE A/C
3197            ;
3198             REDO GOON,1;                        POST MX CODE RESET ROUTINE
3199            ;                                      RESETS NEXT TIME OF FAILURE
3200            ;                                      ALSO CALLED NTOF IN COMMENTS
3201            ;                                      RESETS EACH APPROPRIATE
3202            ;                                      SYSTEM'S ATTRIBUTE
3203                    ACT,,USERF(31).EQ.2.OR.
3204                         USERF(31).EQ.3,RED1;   TO SYSTEM 1 RESET
3205                    ACT,,USERF(32).EQ.2.OR.
3206                         USERF(32).EQ.3,RED2;   TO SYSTEM 2 RESET
3207                    ACT,,USERF(33).EQ.2.OR.
```

```
3208                    USERF(33).EQ.3,RED3;     TO SYSTEM 3 RESET
3209                ACT,,USERF(34).EQ.2.OR.
3210                    USERF(34).EQ.3,RED4;     TO SYSTEM 4 RESET
3211                ACT,,USERF(35).EQ.2.OR.
3212                    USERF(35).EQ.3,RED5;     TO SYSTEM 5 RESET
3213                ACT,,USERF(36).EQ.2.OR.
3214                    USERF(36).EQ.3,RED6;     TO SYSTEM 6 RESET
3215                ACT,,USERF(37).GE.4,DLMT;    STILL NEEDS MAJOR WORK- GO TO MMT
3216                ACT,,,MMRS;                  ELSE, BEGIN ROUTING TO PSO FOR
3217            ;                                  TURNAROUND SERVICE
3218             RED1 ASSIGN,ATRIB(19)=ATRIB(7)+USERF(131),
3219                         ATRIB(18)=USERF(41),1;
3220                ACT,,,RED0;                  SYSTEM 1 RESET
3221            ;
3222             RED2 ASSIGN,ATRIB(20)=ATRIB(7)+USERF(132),
3223                         ATRIB(18)=USERF(42),1;
3224                ACT,,,RED0;                  SYSTEM 2 RESET
3225            ;
3226             RED3 ASSIGN,ATRIB(21)=ATRIB(7)+USERF(133),
3227                         ATRIB(18)=USERF(43),1;
3228                ACT,,,RED0;                  SYSTEM 3 RESET
3229            ;
3230             RED4 ASSIGN,ATRIB(22)=ATRIB(7)+USERF(134),
3231                         ATRIB(18)=USERF(44),1;
3232                  ACT,,,RED0;                  SYSTEM 4 RESET
3233            ;
3234             RED5 ASSIGN,ATRIB(23)=ATRIB(7)+USERF(135),
3235                         ATRIB(18)=USERF(45),1;
3236                ACT,,,RED0;                  SYSTEM 5 RESET
3237            ;
3238             RED6 ASSIGN,ATRIB(24)=ATRIB(7)+USERF(136),
3239                         ATRIB(18)=USERF(46),1;
3240                ACT,,,RED0;                  SYSTEM 6 RESET
3241            ;
```

$w \equiv (70, 74, 78, 82, 86, 90)$ $x \equiv (71, 75, 79, 83, 87, 91)$

$y \equiv (72, 76, 80, 84, 88, 92)$ $z \equiv (73, 77, 81, 85, 89, 93)$

INDIVIDUAL SQ. CONCURRENT SERVICE AREA

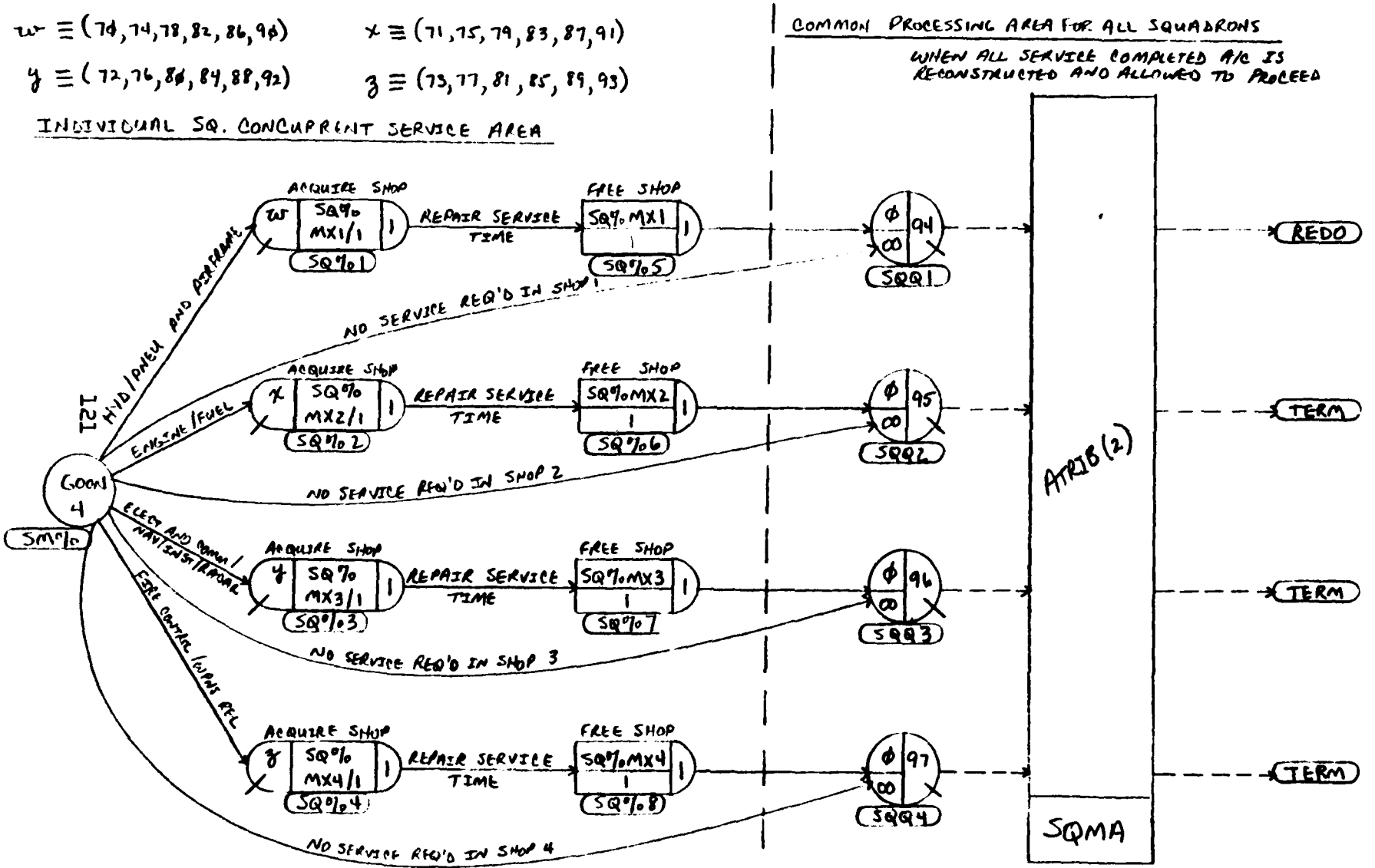
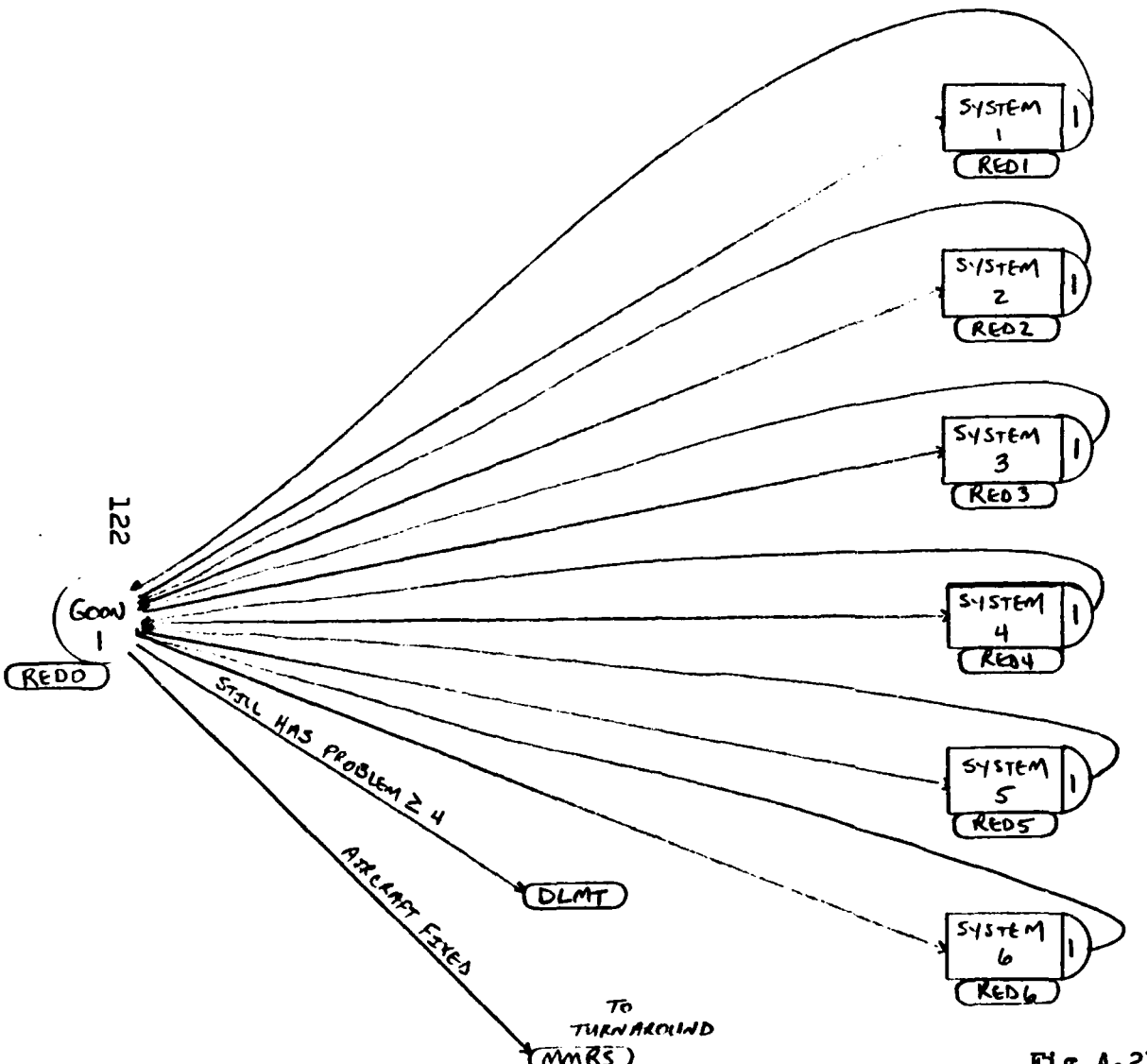


Fig A.21.1 Squadron Maintenance



MECHANISM TO RESET
APPROPRIATE FAIURE
CODES AND NTOF'S

Fig A.21.2 Squadron Maintenance

```
3242          ; EXECUTIVE NETWORK
3243          ;    THE EXECUTIVE NETWORK CONTROLS THE AIRFIELD MODEL.  THE USER
3244          ; INPUTS THE TIMES TO INITIATE KEY ACTIVITIES- SCHEDULER, NIGHT
3245          ; PARKING, QRA CHANGEOVER, AND RESUPPLY AND RECONFIGURATION.
3246          ; THE MASTER CLOCK INITIATES EACH ONE OF THESE ACTIVITIES AT THE
3247          ; USER SPECIFIED TIME.
3248          ;    WHEN SCHEDULER IS INITIATED, THE FIRST THING THAT IS DONE IS
3249          ; INITIALIZATION OF THE FRAG REQUIREMENT FOR THAT DAY.  AFTER
3250          ; INITIALIZATION, THE SCHEDULER BEGINS TO ORGANIZE AND RELEASE
3251          ; MISSIONS.  THIS PROCESS IS CONTINUED UNTIL NIGHTFALL.
3252          ;    NIGHT PARKING TRYS TO PARK ALL A/C IN THE BEST POSSIBLE PARKING
3253          ; SPOT FOR PROTECTION FROM AIR ATTACKS.  THIS MEANS A/C ARE
3254          ; DOUBLED UP AT NIGHT IN THE SHELTERS, EXCEPT THOSE SHELTERS WHICH
3255          ; CONTAIN QRA A/C.
3256          ;    QRA CHANGEOVER IS AS THE NAME IMPLIES. THE PILOTS ON ALERT ARE
3257          ; REPLACED BY FRESH PILOTS.  A/C OF A SQUADRON WHICH HAS BEEN RE-
3258          ; SUPPLIED BY A REPLACEMENT UNIT ARE SWITCHED.
3259          ;    RESUPPLY AND RECONFIGURE CAN ALSO BE INITIATED BY THE MASTER
3260          ; CLOCK.  RESUPPLY DETERMINES IF A SQUADRON REQUIRES A FRESH
3261          ; UNIT BE FLOWN IN TO REPLACE IT.  IF SO, IT SCHEDULES A NEW
3262          ; SQUADRON TO ARRIVE THE FOLLOWING DAY.  RECONFIGURE PREPARES
3263          ; THE A/C FOR THE NEXT DAY'S FRAG.  IT DETERMINES HOW MANY SQUAD-
3264          ; RONS WILL BE CONFIGURED FOR EACH GEOGRAPHIC AREA THE WINGS WILL
3265          ; BE GOING TO.  AFTER RESUPPLY/RECONFIGURATION IS COMPLETED, THE
3266          ; JUNK FILE IS CLEANED OUT TO PREPARE FOR THE FOLLOWING DAY.  A
3267          ; PRINT OUT IS AVAILABLE BY USING LEVEL 1 PRINT OPTION IN THE FORTRAN.
3268          ;    WHEN ALL MAJOR EVENTS HAVE BEEN COMPLETED, OR WHEN THE USER HAS
3269          ; DETERMINED, THE MASTER CLOCK TERMINATES ITSELF AND THE SIMULA-
3270          ; TION RUN COMES TO A STOP.
3271          ;
3272          ;*********************
3273          ; EXECUTIVE NETWORK
3274          ;*********************
3275          ;
3276          ;
3277           EXEC CREATE,,,,1,1;
3278                  ACT,.1;
3279          ;
3280           EXE1 EVENT,1,2;                        MASTER CLOCK
3281                  ACT,XX(98),,EXE1;
3282                  ACT,,XX(96).EQ.1,EXE2;
3283                  ACT,,XX(96).EQ.2,EXE4;
3284                  ACT,,XX(96).EQ.3,EXE6;
3285                  ACT,,XX(96).EQ.4,EXE7;
3286                  ACT,,XX(96).EQ.0,EXTM;
3287          ;
3288           EXE2 EVENT,2,1;
3289                  ACT,,,EXE3;
3290          ;
3291           EXE3 EVENT,3,1;                        SCHEDULER
```

```
3292                    ACT,XX(98),XX(96).EQ.8,EXE3;
3293                  ACT,,XX(96).EQ.1,EXR1;
3294                  ACT,,XX(96).EQ.2,EXR2;
3295                  ACT,,XX(96).EQ.3,EXR3;
3296                  ACT,REL(PSOX),XX(96).EQ.7,EXD2;
3297                  ACT,,XX(96).EQ.4,EXR4;
3298                  ACT,,XX(96).EQ.5,EXR5;
3299                  ACT,,XX(96).EQ.6,EXR6;
3300                  ACT,,XX(96).EQ.0,EXTM;
3301          ;
3302           EXR1 OPEN,RDYPOOL1,1;
3303                  ACT,.0001,,EXE3;
3304          ;
3305           EXR2 OPEN,RDYPOOL2,1;
3306                  ACT,.0001,,EXE3;
3307          ;
3308           EXR3 OPEN,RDYPOOL3,1;
3309                  ACT,.0001,,EXE3;
3310          ;
3311           EXD2 GOON,1;
3312                  ACT,.0016,,EXE3;
3313          ;
3314           EXR4 OPEN,RDYPOOL4,1;
3315                  ACT,.0001,,EXE3;
3316          ;
3317           EXR5 OPEN,RDYPOOL5,1;
3318                  ACT,.0001,,EXE3;
3319          ;
3320           EXR6 OPEN,RDYPOOL6,1;
3321                  ACT,.0001,,EXE3;
3322          ;
3323          ;
3324           EXE4 EVENT,4,1;                        INITIAL NIGHT PARKING
3325          ;
3326           EXE5 EVENT,5,1;
3327                  ACT,REL(PSOX),XX(96).EQ.1,EXD4;
3328                  ACT,,XX(96).EQ.0,EXTM;
3329          ;
3330           EXD4 GOON,1;
3331                  ACT,.0001,,EXE5;
3332          ;
3333          ;
3334           EXE6 EVENT,6,1;                        SWITCH QRA CREWS
3335                  ACT,,,EXTM;
3336          ;
3337           EXE7 EVENT,7,1;                        SET UP RESUPPLY REQUEST FOR
3338          ;                                       NEXT DAY, RECONFIGURE A/C
3339          ;
3340           EXJ1 OPEN,JUNK,1;                      DISPOSE OF JUNK FILE
3341                  ACT,.0001;
```

```
 3342            ;
3343            EXJ2 CLOSE,JUNK,1;
3344            ;
3345            EXTM TERMINATE;
3346            ;
3347            ;
3348                 ENDNETWORK;
3349            ;
3350            INIT,0,4320;
3351            MONTR,TRACE,1400,1600,1,2,46,18,13;
3352            FIN;
```

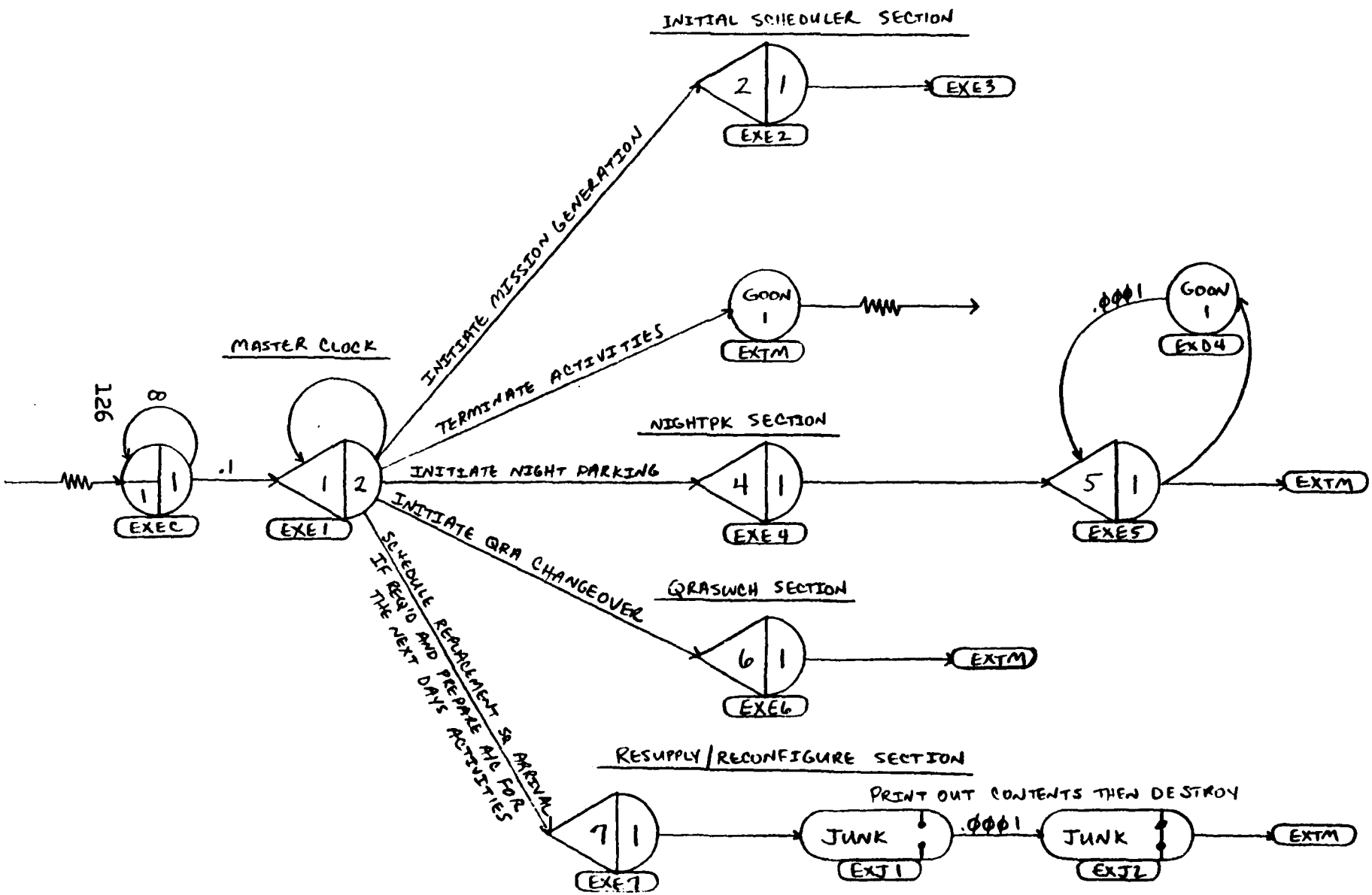


Fig A.22.1 Executive Network

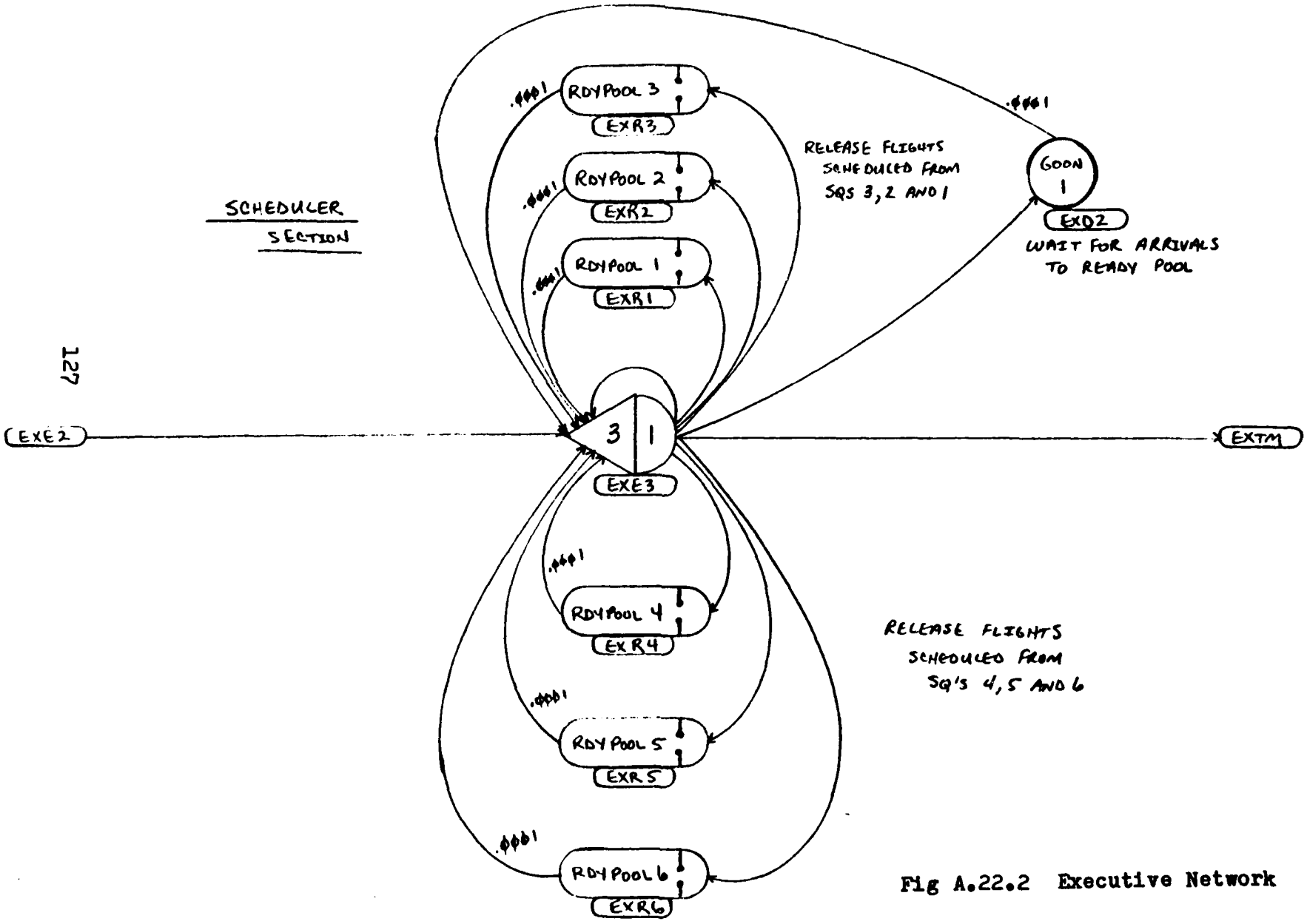


Fig A.22.2 Executive Network

Appendix B: FORTRAN Coding

This appendix contains a complete listing of the FORTRAN code. Extensive comments have been added to the code to help the reader/user understand the purpose and function of each routine. The code is easily transportable and should require no modification for any other machine. However, the code is not independent of the supporting SLAM processor, since many function calls are to SLAM provided routines.

```
C     THE FOLLOWING IS THE MAIN PROGRAM.  THE SIZE OF MEMORY USED TO
C     STORE THE ENTITIES AND SLAM PROGRAM IS SET IN THIS PROGRAM.
C     THE VALUE OF 46000 IS THE SIZE USED FOR THIS SIMULATION RUN.
C
      PROGRAM MAIN(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,TAPE7)
      DIMENSION NSET(46000)
      COMMON/SCOM1/ATRIB(100),DD(100),DDL(100),DTNOW,II,MFA,MSTOP,NCLNR
     &,NCRDR,NPRNT,NNRUN,NNSET,NTAPE,SS(100),SSL(100),TNEXT,TNOW,XX(100)
      COMMON QSET(46000)
      EQUIVALENCE (NSET(1),QSET(1))
      NNSET=46000
      NCRDR=5
      NPRNT=6
      NTAPE=7
      CALL SLAM
      STOP
      END
```

```
      C
      SUBROUTINE USER1
      C
      COMMON/SCOM1/ ATRIB(100),DD(100),DDL(100),DTNOW,II,MFA,MSTOP,NCLNR
     1,NCRDR,NPRNT,NRUN,NSET,NTAPE,SS(100),SSL(100),TNEXT,TNOW,XX(100)
      COMMON/PFLAG/MAXPRT,LEVPRT,BPRT,EPRT
      COMMON/RSCARCE/NUMRES(40)
      C
      C     EVENT COMMON BLOCKS:
      COMMON/MSNREQ /INITAC(3),REQPOT(3),LIMITAC,ACTIVE(6)
      COMMON/SCHEDLR/NGAGLE(3,2),DELAY(3,2),SRATE(3)
      C
      C     USER COMMON BLOCKS:
      COMMON/UCOM1 /LCRSH(24),NCRSH,LTOW(24),NTOW,LEAT(24),NEAT,
     &              PFRCUN2,PFRCUN3,PCUNEXP,PCUNRUN,PBMDROP,PHUNCBM,
     &              PATTR2(3),PATTR3(3),PDAM2(3),PDAM3(3),PDL(5),
     &              DURMIN(3),DURMOD(3),DURMAX(3),PRLCENT(3),PRLWING(3)
      COMMON/UCOM2 /SYS(6,4),SYSTOL(6),MTBF(6),ALP(6),BET(6),XBATREF(5),
     &              ERUNMIN,ERUNMAX
      COMMON/UCOM3 /NODE(7(3),NQ,ING(3),CPCENT,CPWING,CPINTR,
     &              CDRATE,SNDMAX,PERLEFT(3),REFRAT(3),UPTIME,DNTIME,
     &              DARMNOR(3),DARMHUG(3),DARMRUN(3),DARMEXP(3),
     &              RARMNOR(3),RARMHUG(3),RARMRUN(3),RARMBGN(3)
      C
      LOGICAL ACTIVE
      REAL MTBF
      C
      C
```

```
C
C     ESTABLISH FRAC FOR THE THREE DAYS OF THE SIMULATION.
C
C        SORTIE RATES ARE:
         DATA SRATE/3.0,2.0,1.5/
C
C        NUMBER OF MISSIONS PER GAGGLE ON EACH OF TWO GAGGLES TO AREA 3
C        ON EACH OF THE THREE DAYS.
C                       DAY: 1  2  3
C        GAGGLE 1:
              DATA NGAGLE/ 8, 0, 0,
C        GAGGLE 2:
     &                       8, 0, 0/
C
C        DELAY PERIOD AFTER LAUNCHING EACH OF THE GAGGLES:
C                       DAY  1    2    3
C        GAGGLE 1:
              DATA DELAY/20.0,20.0,22.0,
C        GAGGLE 2:
     &                    20.0,20.0,20.0/
C
C        DESIRED PERCENTAGE OF SORTIES TO EACH AREA.
C
C                          AREA: 1    2    3
                    DATA REQPCT/0.50,0.30,0.20/
C
C
C     INITIAL NUMBER OF A/C TO BE CONFIGURED FOR EACH AREA.
C     THE SUM MUST EQUAL TOTAL A/C MINUS NUMBER ON GRA.
C
C        A/C CONFIGURED FOR AREA: 1  2  3
                    DATA INITAC/ 0,18,60/
C
C
C
C
```

```
      C     NUMBER OF CENTER LINE AND WING TANKS FOR EACH CONFIGURATION
      C     ARE ESTABLISHED FOR EACH AREA.
      C                 CONFIGURATION    1  2  3
      C        CENTER LINE:
                          DATA NOCENT/ 1, 1, 1/
      C        WING:
                          DATA NOWING/ 0, 2, 2/
      C
      C     LOADING AND UNLOADING TIMES OF EXTERNAL TANKS:
            DATA UPTIME,DNTIME/ 15.0,10.0/
      C
      C
      C     MISSION DURATION TIMES (SORTIE LENGTH AVERAGE)
      C     BY AREA   :     1    2     3
            DATA DURMIN/ 45.0,70.0,80.0/
            DATA DURMOD/ 55.0,80.0,90.0/
            DATA DURMAX/ 70.0,90.0,100.0/
      C
      C
      C     PROBABILITIES USED TO DETERMINE THE STATUS OF THE
      C     A/C DURING THE MISSION.
      C
      C
      C     A/C ATTRITION PROBABILITIES:
      C                 ATTRITION AREA 1    2    3
      C        IN 2-SHIP:
                        DATA PATTR2/ 0.05,0.06,0.07/
      C        IN 3-SHIP:
                        DATA PATTR3/ 0.04,0.05,0.06/
      C
      C
      C     TANK JETTISON PROBABILITIES BY AREA: 1    2    3
      C        CENTER LINE TANKS:
                              DATA PR_CENT/0.40,0.50,0.60/
      C        WING TANKS:
                              DATA PR_WING/1.00,1.00,1.00/
      C
      C
      C
      C     MEAN TIME BETWEEN FAILURE OF SIX SYSTEMS:
            DATA MTBF/5400.0,640.0,900.0,420.0,6000.0,1200.0/
      C     ALPHA FOR THE BETA DISTRIBUTION:
            DATA ALP/5.0,4.0,4.0,3.0,5.0,4.0/
      C     BETA FOR THE BETA DISTRIBUTION:
            DATA BET/1.5,2.0,2.0,1.5,1.5,2.0/
      C
      C
      C     SPECIFY IF A SQUADRON IS ACTIVE OR NOT FOR THE SIMULATION.
            ACTIVE(1) = .TRUE.
            ACTIVE(2) = .TRUE.
```

```
      ACTIVE(3) = .TRUE.
      ACTIVE(4) = .TRUE.
      ACTIVE(5) = .TRUE.
      ACTIVE(6) = .TRUE.
C
C     SPECIFY NUMBER OF A/C REQUIRED FOR A SQUADRON TO BE CONSIDERED
C     OPERATIONAL.  USED TO DETERMINE IF A SQUADRON REQUIRES RESUPPLY.
      LIMITAC = 12
C
C     INITIAL PERCENT OF A/C OPERATIONAL:
      XX(64) = .90
C
C     NUMBER OF PILOTS IN EACH SQUADRON:
      XX(58) = 22.0
C
C     NUMBER OF AIRCRAFT IN EACH SQUADRON:
      XX(57) = 16.0
C
C     NUMBER OF PILOTS ON QRA ALERT:
      XX(61) = 3
C
C     NUMBER OF PILOTS QRA QUALIFIED (INCLUDE THOSE ON ALERT):
      XX(62) = 9
C
C     NUMBER OF PILOTS FLIGHT LEAD QUALIFIED (INCLUDE THOSE ON
C     ALERT AND THOSE QRA QUALIFIED, BUT DO NOT COUNT TWICE):
      XX(63) = 14
C
C     INITIAL POL SUPPLY:
      XX(60) = 1000000.0
```

```
170             C
171             C     DETERMINE THE NUMBER OF RESOURCES FOR EACH OF THE FOUR
172             C     WING SHOPS:
173             C
174                   NUMRES(1) = 2
175                   NUMRES(2) = 2
176                   NUMRES(3) = 2
177                   NUMRES(4) = 2
178             C
179             C
180             C     INITIALIZE THE NUMBER OF RESOURCES FOR EACH OF THE
181             C     SIX MMT UNITS:
182             C
183                   NUMRES(5) = 2
184                   NUMRES(6) = 2
185                   NUMRES(7) = 2
186                   NUMRES(8) = 2
187                   NUMRES(9) = 2
188                   NUMRES(10) = 2
189             C
190             C
191             C     ESTABLISH THE NUMBER OF RESOURCES FOR EACH OF THE FOUR
192             C     FOUR SHOPS IN EACH OF THE SIX SQUADRONS:
193             C
194             C     SQUADRON 1 -
195                      NUMRES(12) = 4
196                      NUMRES(13) = 4
197                      NUMRES(14) = 4
198                      NUMRES(15) = 4
199             C
200             C     SQUADRON 2 -
201                      NUMRES(16) = 4
202                      NUMRES(17) = 4
203                      NUMRES(18) = 4
204                      NUMRES(19) = 4
205             C
206             C     SQUADRON 3 -
207                      NUMRES(20) = 4
208                      NUMRES(21) = 4
209                      NUMRES(22) = 4
210                      NUMRES(23) = 4
211             C
212             C     SQUADRON 4 -
213                      NUMRES(24) = 4
214                      NUMRES(25) = 4
215                      NUMRES(26) = 4
216                      NUMRES(27) = 4
217             C
218             C     SQUADRON 5 -
219                      NUMRES(28) = 4
```

```
                  NUMRES(29) = 4
                  NUMRES(30) = 4
                  NUMRES(31) = 4
C
C     SQUADRON 6 -
                  NUMRES(32) = 4
                  NUMRES(33) = 4
                  NUMRES(34) = 4
                  NUMRES(35) = 4
C
C
C     INITIAL NUMBER OF MX TEAMS:
                  NUMRES(11) = 78
C
C     INITIAL NUMBER OF REARMING CREWS:
                  NUMRES(36) = 18
C
C     INITIAL NUMBER OF FUEL TRUCKS:
                  NUMRES(37) = 40
C
C     INITIAL NUMBER OF DEARMING CREWS:
                  NUMRES(38) = 6
C
C     NUMBER OF RUNWAYS:
                  NUMRES(39) = 1
C
C     NUMBER OF HOTPITS:
                  NUMRES(40) = 4
C
C
C
C
C
C     SWITCH TO CONTROL TRACING PRINT STATEMENTS
C        0   -   NONE
C        1   -   JUNK A/C FILE
C        2   -   MAJOR EVENTS BEING CALLED
C        3   -   DETAILS OF MAJOR EVENTS
C        4   -   ALL PRINTS, EXCEPT FUNCTION USERF AND SUBR EVENT
C        5   -   ALL PRINTS, EXCEPT FUNCTION USERF
C        6   -   ALL PRINTS
C
                  LEVPRT    =    1
C
C     NOW WHEN TO START AND STOP PRINTING STATEMENTS:
                  BPRT = -1.0
                  EPRT = -1.0
C
C     STATE NUMBER OF FILES TO BE DUMPED:
                  MAXPRT = 0
```

```
270                C
271                      RETURN
272                      END
273          C
274          C
275          C
```

```
      SUBROUTINE INTLC
      COMMON/SCOM1/ ATRIB(100),DD(100),DDL(100),DTNOW,II,MFA,MSTOP,NCLNR
     &,NCRDR,NPRNT,NRUN,NNSET,NTAPE,SS(100),SSL(100),TNEXT,TNOW,XX(100)
      COMMON/PFLAG/MAXPRT,LEVPRT,BPRT,EPRT
      COMMON/RSORCE/NUMRES(40)
C
C     EVENT COMMON BLOCKS:
      COMMON/STATS  /MAXCONF(6),NACTYPE(6,3),NCSQ(300),MSNRQA2,MSNFLW(3)
      COMMON/MSNREQ /INITAC(3),REQPOT(3),LIMITAC,ACTIVE(6)
      COMMON/SCHEDLR/NGAGLE(3,2),DELAY(3,2),SRATE(3)
      COMMON/CLOCK  /INTDARK,DUSK(3),DAYLGHT(3),MAJEVNT(13,2)
C
C     USER COMMON BLOCKS:
      COMMON/UCOM1 /LCRSH(24),NCRSH,LTOW(24),NTOW,LBAT(24),NBAT,
     &              PERGUN2,PERGUN3,PGUNEXP,PGUNRUN,PBMDROP,PHUNGBM,
     &              PATTR2(3),PATTR3(3),PDAM2(3),PDAM3(3),PDL(5),
     &              DURMIN(3),DURMOD(3),DURMAX(3),PRLCENT(3),PRLWING(3)
      COMMON/UCOM2 /KP4RK(6,50,2),NTYPE(6,3)
      COMMON/UCOM5 /SYS(6,4),SYSTOL(6),MTBF(6),ALP(6),BET(6),NBATREP(5),
     &              ERUNMIN,ERUNMAX
      COMMON/UCOM6 /DIST(13,13),TOW(3),TAXI(3),CREW(3)
      COMMON/UCOM7 /NOCENT(3),NOWING(3,),CPCENT,CPWING,CPINTR,
     &              ODRATE,GNDMAX,PERLEFT(3),REFRAT(3),UPTIME,DNTIME,
     &              DARMNOR(3),DARMDUG(3),DARMRUN(3),DARMEXP(3),
     &              RARMNOR(3),RARMDUG(3),RARMCUN(3),RARMBGN(3)
      COMMON/UCOM8 /MIN1(5),MODE1(5),MAX1(5),MIN2(5),MODE2(5),MAX2(5),
     &              MIN3(5),MODE3(5),MAX3(5),MIN4(5),MODE4(5),MAX4(5),
     &              MIN5(5),MODE5(5),MAX5(5),MIN6(5),MODE6(5),MAX6(5),
     &              WSH1INT(3),WSH3INT(3)
      COMMON/UCOM9 /MMTMIN(6,2),MMTMOD(6,2),MMTMAX(6,2)
      COMMON/UCOM10/MINI(5),MODEI(5),MAXI(5)
C
      REAL MIN1,MODE1,MAX1,MIN2,MODE2,MAX2,MIN3,MODE3,MAX3,
     &     MIN4,MODE4,MAX4,MIN5,MODE5,MAX5,MIN6,MODE6,MAX6,
     &     MINI,MODEI,MAXI,MMTMIN,MMTMOD,MMTMAX,MTBF,MAJEVNT

      LOGICAL INTDARK,ACTIVE
C

      DICTIONARY OF VARIABLES.


        NAVAIL - TOTAL NUMBER A/C AVAILABLE
        LCRSH  - LIST OF ENCODED SYSTEM LEVELS WHICH CAN CAUSE THE A/C
                 TO CRASH (NOT INCLUDING BATTLE DAMAGE).
        NCRSH  - NUMBER OF ENCODED VALUES IN LCRSH (LENGTH OF LIST).
        LTOW   - LIST OF ENCODED SYSTEM LEVELS WHICH CAUSE THE A/C TO
                 REQUIRE TOWING.
        NTOW   - LENGTH OF LIST LTOW.
        LBAT   - LIST OF ENCODED VALUES USED TO DETERMINE IF THE A/C
```

```
C                CRASHED DUE TO MX FAILURES AND BATTLE DAMAGE
C       NBAT   - LENGTH OF LIST LBAT.
C
C       NTYPE  - NUMBER OF EACH TYPE OF PARKING SPACE FOR EACH BDDN.
C
C       SYS    - IF SYSTEM FAILS, CUMULATIVE PROBABILITIES FOR EACH
C                SYSTEM USED TO DETERMINE LEVEL OF FAILURE (LEVEL 5
C                BEING 1.00).
C       SYSTOL - TOLERANCE WHICH ALLOWS MX FAILURES TO OCCUR IF THE
C                SYSTEM IS SCHEDULED TO FAIL WITHIN SYSTOL MINUTES.
C       MTBF   - MEAN TIME BETWEEN FAILURE
C
C
C       DIST   - DISTANCE MATRIX FOR POINT I TO POINT J ON BASE.
C       TOW    - LOW,MEDIUM,HIGH INPUTS TO TRIANGULAR DISTRIBUTION TO
C                DETERMINE A/C TOW RATE.
C       TAXI   - AS ABOVE, EXCEPT TO DETERMINE A/C TAXI RATE.
C       CREW   - AS ABOVE, EXCEPT TO DETERMINE CREW RETURN RATE FROM
C                VARIOUS AREAS OF THE BASE AFTER GND ABORT OR MX
C                FAILURE.
C
C       GDRATE - POL CONSUMPTION RATE ON THE GROUND.
C       CPCENT - CAPACITY OF CENTERLINE TANK(S)
C       CPWING - CAPACITY OF EXTERNAL WING TANKS
C       CPINTR - CAPACITY OF INTERNAL TANK(S).
C       GNDMAX - MAXIMUM GROUND OPERATING TIME BEFORE TOPPING OFF TANKS
C       UPTIME - TIME TO UPLOAD ONE EXTERNAL TANK.
C       DNTIME - TIME TO DOWN-LOAD AN EXTERNAL TANK.
C
C     END OF GLOSSARY.
C
C
C
C
C
C
C     SERVICE TIMES FOR SYSTEMS 1 TO 6:
C               LEVELS      1    2    3    4    5
C
C     SYSTEM 1:
                DATA MIN1 /15.0,20.0,25.0,60.0,420.0/
                DATA MODE1/20.0,25.0,45.0,180.0,540.0
                DATA MAX1 /30.0,35.0,60.0,360.0,660.0/
C     SYSTEM 2:
                DATA MIN2 /25.0,30.0,120.0,720.0,720.0/
                DATA MODE2/75.0,120.0,130.0,1150.0,1200.0/
                DATA MAX2 /120.0,240.0,300.0,1700.0,1800.0/
C     SYSTEM 3:
                DATA MIN3 /40.0,60.0,120.0,360.0,360.0/
                DATA MODE3/75.0,180.0,270.0,800.0,900.0/
                DATA MAX3 /120.0,240.0,360.0,1100.0,1200.0/
```

```
C     SYSTEM 4:
      DATA MIN4 /30.0,60.0,180.0,270.0,420.0/
      DATA MODE4/45.0,90.0,270.0,600.0,900.0/
      DATA MAX4 /60.0,120.0,420.0,900.0,1200.0/
C     SYSTEM 5:
      DATA MIN5 /30.0,40.0,60.0,90.0,180.0/
      DATA MODE5/40.0,60.0,80.0,120.0,300.0/
      DATA MAX5 /50.0,90.0,100.0,240.0,420.0/
C     SYSTEM 6:
      DATA MIN6 /40.0,60.0,90.0,120.0,150.0/
      DATA MODE6/50.0,120.0,180.0,180.0,210.0/
      DATA MAX6 /60.0,180.0,240.0,300.0,360.0/
C
C
C
C     WING SHOP SERVICE TIME INCREASE DUE TO BOTH PROBLEMS BEING IN SAME
C     SHOP.  ADDED TO MAXIMUM SERVICE TIME OF THE TWO SERVICES AND THE
C     RESULT IS USED AS THE OVERALL SERVICE TIME.
C                    MIN  MODE  MAX
      DATA WSH1INT/30.0,50.0,60.0/
      DATA WSH3INT/25.0,35.0,45.0/
C
C
C
C
C
C     MMT SERVICE TIMES:
C
C                          SYSTEMS: 1    2     3     4     5    6
C       LEVEL 4 PROBLEM:
      DATA (MMTMIN(I,1),I=1,6)/70.00,100.0,70.00,70.00,55.0,55.0/
      DATA (MMTMOD(I,1),I=1,6)/100.0,130.0,120.0,100.0,70.0,70.0/
      DATA (MMTMAX(I,1),I=1,6)/140.0,180.0,170.0,130.0,85.0,85.0/
C
C       LEVEL 5 PROBLEM:
      DATA (MMTMIN(I,2),I=1,6)/100.0,130.0,120.0,130.0,65.00,70.00/
      DATA (MMTMOD(I,2),I=1,6)/130.0,150.0,180.0,190.0,95.00,90.00/
      DATA (MMTMAX(I,2),I=1,6)/200.0,200.0,260.0,260.0,130.0,120.0/
C
C
C     INTERFERANCE TIMES ADDED TO SQUADRON SERVICE TIMES DUE
C     TO OTHER SHOPS WORKING THE SAME A/C AT THE SAME TIME.
C
C     NUMBER OF OTHER ACTIVITIES: 1   2   3    4    5
                    DATA MINI /5.0,5.0,10.0,15.0,20.0/
                    DATA MODEI/5.1,5.1,10.1,20.0,25.0/
                    DATA MAXI /5.2,5.2,10.2,25.0,30.0/
C
C
C     SECTION 1, UTILITY.
```

```
C
C
C
C     LCRSH, LTOW, AND LBAT ARE CODES WHICH DETERMINE IF THE A/C
C     SHOULD CRASH, BE TOWED, OR DESTROYED DUE TO A COMBINATION OF
C     MX FAILURES (WITH LBAT,  BATTLE DAMAGE IS ALSO CONSIDERED).
C     IF EACH SYSTEM HAS A GREATER THAN OR EQUAL LEVEL OF FAILURE
C     THAN THAT WHICH IS SPECIFIED IN ONE OF THE CODES IN LCRSH,
C     LTOW, AND LBAT, THEN THE A/C HAS MEET THE CONDITIONS AND IS
C     CRASHED, TOWED, OR DESTROYED, RESPECTIVELY.
C
C     EACH CODE VECTOR CAN HAVE UP TO 24 DIFFERENT CODED NUMBERS
C     TO COMPARE TO THE MX FAILURE CODE, ATTRIB(18), TO DETERMINE
C     THE OUTCOME OF THE A/C (IE CRASH, TOW, OR DESTROYED).  THE
C     NUMBER OF ENCODED NUMBERS FOR EACH VECTOR IS SPECIFIED IN THE
C     FOLOWING STATEMENT(THESE NUMBERS ARE FOR THIS PARTICULAR RUN).
      DATA NCRSH,NTOW,NBAT/6,8,13/
C
C     IF LESS THAN 24 ENCODED NUMBERS ARE USED IN THE VECTORS LCRSH,
C     LTOW, AND LBAT; USE 999999 FOR THE REST (9999999 FOR LBAT).
C
      DATA LCRSH/050000,444000,005000,000500,534000,433000,18*999999/
C
      DATA LTOW /500000,050000,040000,005000,000500,000400,
     +           000005,333300,16*999999/
C
C
C     FOR LBAT, THE SEVENTH DIGIT REFERS TO BATTLE DAMAGE.
      DATA LBAT/0500000,4440000,0400005,0050000,0040005,0340004,
     +          0005000,0334004,5000003,5340000,0024005,4330000,
     +          4230004,11*9999999/
C
C
C
C
C     GUN PROBABILITIES:
C        PROBABILITY OF FIRING A GUN IF IN TWO SHIP OR THREE SHIP:
         DATA PFRGUN2,PFRGUN3/ 0.70,0.70/
C
C        PROBABILITY OF THE GUN EXPLODING:
         DATA PGUNEXP/0.01/
C
C        PROBABILITY OF THE GUN RUNNING AWAY:
         DATA PGUNRUN/0.02/
C
C
C
C     BOMB PROBABILITIES:
C
```

```
C       PROBABILITY OF DROPPING THE BOMBS:
      DATA PBMDROP/1.00/
C
C     PROBABILITY OF THE BOMB(S) BEING HUNG:
      DATA PHUNGBM/0.05/
C
C
C
C     BATTLE DAMAGE PROBABILITIES:
C                          AREA: 1    2    3
C       IN 2-SHIP:
              DATA PDAM2 / 0.03,0.02,0.04/
C       IN 3-SHIP:
              DATA PDAM3 / 0.02,0.01,0.03/
C
C     BATTLE DAMAGE LEVEL PROBABILITIES:
C       DAMAGE LEVEL: 1    2    3    4    5
          DATA PDL/0.60,0.20,0.10,0.06,0.04/
C
C
C
C     SECTION 2, PARKING.
C
C
C     SPECIFY THE NUMBER OF EACH TYPE OF PARKING SPACE AVAILABLE
C     (SHELTERED, REVETMENT, DISPERSED) FOR EACH SQUADRON.
C               SQUADRON:   1   2   3   4   5   6
C       SHELTERED:
              DATA NTYPE/ 7,  7,  7,  7,  7,  7,
C       REVETMENT:
     &                    9,  9,  9,  0,  0,  0,
C       DISPERSED:
     &                   34, 34, 34, 43, 43, 43/
C
C
C
C     SECTION 5, ALTER CODE.
C
C
C     CUMULATIVE PROBABILITY OF FAILURE LEVEL FOR THE SIX DIFFERENT
C     MAINTENANCE SYSTEMS.
C               SYSTEMS:  1    2    3    4    5    6
C       LEVEL 1:
            DATA SYS/0.10,0.10,0.05,0.05,0.10,0.05,
C       LEVEL 2:
     &                0.20,0.50,0.35,0.85,0.35,0.50,
C       LEVEL 3:
     &                0.50,0.80,0.50,0.90,0.65,0.75,
C       LEVEL 4:
     &                0.80,0.90,0.65,0.95,0.90,0.90/
```

```
C       SINCE LEVEL FIVE IS 1.00, IT DOES NOT HAVE TO BE SPECIFIED.
C
C
C     TOLERANCE ALLOWED BEFORE A SYSTEM IS GOING TO BREAK AND
C     KNOW.  USED IN PREFLIGHT TO CHECK IF A SYSTEM IS ABOUT
C     TO FAIL.
C        SYSTEM:     1   2   3   4   5   6
         DATA SYSTOL/5.0,5.0,5.0,5.0,5.0,5.0/
C
C
C
C     MX FAILURE CODES THAT ARE EQUIVILENT TO THE FIVE BATTLE
C     DAMAGE LEVELS.  AT MX, THE MX FAILURE CODE, ATRIB(16), AND
C     THE EQUIVILENT BATTLE DAMAGE CODE GIVEN BELOW ARE "MASHED"
C     TOGETHER TO FORM THE NEW MX FAILURE CODE TO ESTABLISH
C     THE LEVEL OF REPAIR TO BE PERFORMED BY MX.
C     999999 INDICATES THAT THE A/C IS NOT REPAIRABLE.
C
      DATA NBATREP/ 122211,233321,344321,999999,999999/
C
C
C
C     MINIMUM AND MAXIMUM VALUES OF UNIFORM DISTRIBUTION
C     USED TO INITIALIZE THE A/C ENGINE RUNNING TIME.
         DATA ERUNMIN,ERUNMAX/30.0,12000.0/
C
C
C
C     SECTION 6, TRAVEL TIME.
C
C     THE FOLLOWING MATRIX DEFINES THE DISTANCES BETWEEN THE THIRTEEN
C     ENCODED LOCATIONS ON THE FIELD.  IF 0.0 IS ENTERED, THE DISTANCE
C     BETWEEN THOSE TWO POINTS IS NOT IMPORTANT TO THE SIMULATION.
C     THE FOLLOWING IS A LIST OF THE ENCODED LOCATIONS:
C        1   SQUADRON AREA 1
C        2   SQUADRON AREA 2
C        3   SQUADRON AREA 3
C        4   SQUADRON AREA 4
C        5   SQUADRON AREA 5
C        6   SQUADRON AREA 6
C        7   ARMING AREA
C        8   APPROACH END OF THE RUNWAY
C        9   WING AREA
C        10  DEARMING AREA
C        11  HOTPIT AREA
C        12  DEPARTURE END OF THE RUNWAY
C        13  NOT DEFINED
C
C     DIST. MATRIX: 1   2   3   4   5   6   7   8   9   10  11  12  13
      DATA DIST   /0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.5,1.0,0.5,2.0,0.7,2.3,0.0,
```

```
     +            0.0,2.0,0.0,0.0,0.0,0.0,0.0,..3,2.0,0.5,0.0,0.7,1.2,0.0,
     +           0.0,0.0,0.0,0.0,2.0,0.0,0.0,1.7,2.0,0.0,0.7,0.0,1.1,0.0,0.0,
     +           0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.5,1.4,1.1,1.4,5.0,0.0,
     +           0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,1.0,1.0,1.9,5.5,1.9,5.5,0.0,
     +           0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,1.0,1.5,2.4,4.0,2.4,4.0,0.0,
     +           0.5,1.0,1.7,0.0,1.0,1.0,0.0,2.4,1.0,2.0,0.0,0.0,2.0,0.0,
     +           1.0,2.0,2.0,0.5,1.0,1.5,2.4,0.0,1.0,2.4,1.0,2.4,0.0,
     +           0.5,0.5,0.0,1.4,1.5,2.4,1.0,1.0,0.0,1.2,0.5,1.2,0.0,
     +           2.0,0.0,0.0,3.0,3.5,4.0,2.0,2.4,1.2,0.0,1.5,0.0,0.0,
     +           0.7,0.7,1.1,1.4,1.9,2.4,0.0,1.0,0.5,1.0,0.0,1.0,0.0,
     +           1.0,1.2,0.0,0.0,3.0,3.5,4.0,2.0,2.4,1.2,0.0,1.5,0.0,0.0,
     +           0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0,0.0/
C
C
C    LOW, MEDIUM, AND HIGH RATES OF TRAVEL (SPEED) THAT AN A/C CAN
C    TAXI, THAT AN A/C CAN BE TOWED, AND THAT A CREW CAN BE BUSED ON
C    AN AIRFIELD.
      DATA TOW /.134,.268,.402/
      DATA TAXI/.402,.671,1.073/
      DATA CREW/.671,.939,1.073/
C
C
C    SECTION 7, TURNAROUND
C
C
C
C    FUEL CAPACITY OF CENTER LINE TANK, WING TANK, AND INTERNAL TANKS:
      DATA CPCENT,CPWING,CPINTR/600.0,800.0,5600.0/
C
C    FUEL CONSUMPTION RATE ON THE GROUND:
      DATA GDRATE/40.0/
C
C    MAX GROUND OPERATING TIME BEFORE REQUIRING REFUELING:
      DATA GNDMAX/45.0/
C
C    PERCENT OF INTERNAL FUEL LEFT AFTER A MISSION:
C              AREA   1    2    3
      DATA PERLEFT/ .10,.15,.20/
C
C    REFUELING RATE:  MIN   MODE  MAX
      DATA REFRAT/ 440.0,500.0,560.0/
C
C    REARMING TIMES:
C                                    MIN MOD MAX
C       NORMAL:
                  DATA DARMNOR/ 1.0,2.0,3.0/
C       HUNG ORD.:
                  DATA DARMHUG/ 4.0,6.0,10.0/
C       RUNWAY GUN:
                  DATA DARMRUN/ 5.0,9.0,15.0/
```

```
626              C       EXPLODED GUN:
627                              DATA DARMEXP/ 5.0,8.0,20.0/
628              C
629              C
630              C     REARMING TIMES:
631              C                                      MIN   MOD   MAX
632              C        NO ORDINANCE:
633                              DATA RARMNOR/ 12.0,15.0,24.0/
634              C        UNLOADED GUN:
635                              DATA RARMGUN/  5.0,10.0,15.0/
636              C        HUNG ORD.:
637                              DATA RARMHUG/  5.0,10.0,15.0/
638              C        BAD GUN:
639                              DATA RARMBGN/ 60.0,120.0,180.0/
640              C
641              C
642              C
643              C
644              C     ESTABLISH, RELATIVE TO TNOW, WHAT TIME IT BECOMES DAY LIGHT,
645              C     WHAT TIME IT GETS DARK, AND WHETHER IT IS INITIALLY DARK (INITDARK)
646              C                    DAY   1     2     3
647                     DATA DAY_LHT/ 10.0,1455.0,2895.0/
648                     DATA DUSK   /975.0,2415.0,3855.0/
649              C
650                     INITDARK = .TRUE.
651              C
652              C
653              C     MAJEVNT IS THE VECTOR THAT DRIVES THE MASTER CLOCK.  IT DETERMINES
654              C     WHEN KEY ACTIVITIES ARE TO BEGIN.  THE KEY ACTIVITIES ARE:
655              C        0   TERMINATE THE RUN
656              C        1   START SCHEDULING FLIGHTS
657              C        2   BEGIN NIGHT PARKING OF A/C
658              C        3   PERFORM QRA CHANGEOVER
659              C        4   DETERMINE RESUPPLY REQUIREMENT AND RECONFIGURE FOR NEXT DAY
660              C        99  NOTHING
661              C     THE FIRST VALUE OF EACH PAIR IS THE TIME RELATIVE TO THE BEGINNING
662              C     OF THE SIMULATION THAT THE ACTIVITY SPECIFIED BY THE SECOND VALUE
663              C     OF THE PAIR IS TO BE INITIATED (IE. THE FIRST PAIR OF NUMBERS IS
664              C     10.0 AND 4.0).  A MAXIUM OF THIRTEEN ACTIVITIES CAN BE SCHEDULED.
665              C
666                     DO 13 I = 1,13
667                        MAJEVNT(I,1) = 999999.0
668                        MAJEVNT(I,2) = 99.0
669                13   CONTINUE
670              C
671                     MAJEVNT(1,1) = 10.0
672                     MAJEVNT(1,2) = 4.0
673                     MAJEVNT(2,1) = 16.0
674                     MAJEVNT(2,2) = 1.0
675                     MAJEVNT(3,1) = 976.0
```

```
676                MAJEVNT(3,2) = 3.0
677                MAJEVNT(4,1) = 980.0
678                MAJEVNT(4,2) = 2.0
679          C
680                MAJEVNT(5,1) = 1450.0
681                MAJEVNT(5,2) = 4.0
682                MAJEVNT(6,1) = 1456.0
683                MAJEVNT(6,2) = 1.0
684                MAJEVNT(7,1) = 2416.0
685                MAJEVNT(7,2) = 3.0
686                MAJEVNT(8,1) = 2420.0
687                MAJEVNT(8,2) = 2.0
688          C
689                MAJEVNT(9,1) = 2890.0
690                MAJEVNT(9,2) = 4.0
691                MAJEVNT(10,1)= 2896.0
692                MAJEVNT(10,2)= 1.0
693                MAJEVNT(11,1)= 3856.0
694                MAJEVNT(11,2)= 3.0
695                MAJEVNT(12,1)= 3860.0
696                MAJEVNT(12,2)= 2.0
697          C
698                MAJEVNT(13,1)= 4319.0
699                MAJEVNT(13,2)= 0.0
700          C
701          C
702          C     ZERO OUT THE GLOBAL VARIABLES.
703                DO 100 I = 1,100
704                   XX(I) = 0.0
705           100  CONTINUE
706          C
707          C     CALL THE USER INITIALIZATION ROUTINE AND SET THE RESOURCE LEVELS
708          C
709                CALL USERI
710          C
711                DO 200 I = 1,40
712                   CALL ALTER(I,N,MRES(I))
713           200  CONTINUE
714          C
715          C
716          C     PROBABILITY OF A/C DELAY AT PILOT PREFLIGHT:
717                   XX(65) = .25
718          C
719          C     PROBABILITY OF A/C DELAY AT START ENGINE:
720                   XX(66) = .15
721          C
722          C     PROBABILITY OF A/C DELAY AT TAXI,MARSHALL,ARM:
723                   XX(67) = .10
724          C
725          C     PROBABILITY OF FLIGHT DELAY AT TAKE-OFF:
```

```
726                   XX(68) = .03
727             C
728             C     PROBABILITY OF A/C DELAY AT REJOIN:
729                   XX(69) = .10
730             C
731             C
732             C     ZERO OUT PARKING AREA.
733             C
734                   DO 300 I = 1,6
735                   DO 300 J = 1,50
736                   DO 300 K = 1,2
737                     NPARK(I,J,K) = 0
738              300  CONTINUE
739             C
740             C     MARK IRA PARKING SPACES.
741                   DO 310 I = 1,6
742                   DO 310 J = 1,IFIX(XX(61))
743                   NPARK(I,J,2) = 1000000
744              310  CONTINUE
745             C
746             C
747                   DO 678 I = 1,6
748                     IF(ACTIVE(I))XX(49 + I) = 1.0
749              678  CONTINUE
750             C
751             C
752             C     SPECIFY INITIAL CONFIGURATION OF RESUPPLY A/C:
753                     XX(59) = 2
754             C
755                   DO 500 I = 1,300
756                     NCSD(I) = 999999
757              500  CONTINUE
758             C
759                   RETURN
760                   END
761             C
762             C
763             C
764             C
765             C
766             C
767             C
768             C
```

```
 769                SUBROUTINE OTPUT
 770               COMMON/SCOM1/ATRIB(100),DD(100),DDL(100),DTNOW,II,MFA,MSTOP,NCLNR
 771              &,NCRDR,NPRNT,NNRUN,NNSET,NTAPE,SS(100),SSL(100),TNEXT,TNOW,XX(100)
 772               COMMON/PFLAG/MAXPRT,LEVPRT,BPRT,EPRT
 773         C
 774         C     SECTION TO PRINT A FILE DUMP AT THE END OF A RUN.
 775         C
 776               IF(MAXPRT.GT.99)MAXPRT = 99
 777               IF(MAXPRT.GT.0)THEN
 778                  DO 100 I = 1,MAXPRT
 779                     CALL PRNTF(I)
 780          100     CONTINUE
 781               ENDIF
 782         C
 783               IF(LEVPRT.GE.2.AND.TNOW.GE.BPRT.AND.TNOW.LE.EPRT)THEN
 784                  PRINT*,'***  NUMBER OF A/C FAILING ',XX(100)
 785               ENDIF
 786         C
 787               PRINT*,'-------------------'
 788               PRINT*,
 789               PRINT*,'*RESPONSE VARIABLE*'
 790               PRINT*,XX(94)
 791               PRINT*,
 792               PRINT*,'-------------------'
 793               RETURN
 794               END
 795         C
 796         C
 797         C
 798         C
 799         C
 800         C
 801         C
 802         C
 803         C
```

```
      FUNCTION USERF(IFN)
      COMMON/SCOM1/ ATRIB(100),DD(100),DD_(100),DTNOW,II,MFA,MSTOP,NCLNR
     &,NCRDR,NPRNT,NNRUN,NNSET,NTAPE,SS(100),SSL(100),TNEXT,TNOW,XX(100)
      COMMON/PFLAG/MAXPRT,LEVPRT,BPRT,EPRT
      COMMON/RSORCE/NUMRES(40)
      COMMON/STATS/MAXCONF(6),NACTYPE(6,3),NCSQ(300),MSNRQAZ,MSNFLW(3,
C
C
C     ALL THE USER FUNCTIONS (USERF) ARE DIVIDED INTO THIRTEEN SECTIONS:
C
C     SECTION     SECTION        ENTRY     FUNCTION NUMBER      NUMBER
C     NUMBER       NAME         ADDRESS      RANGE (IFN)        IN USE
C     ------      -------       -------    ---------------      ------
C        1        UTILITY         1000        11  TO   19         9
C        2        PARKING         2000        21  TO   29         3
C        3        EXAMINE CODE    3000        31  TO   39         9
C        4        RESET CODE      4000        41  TO   49         7
C        5        ALTER CODE      5000        51  TO   59         3
C        6        TRAVEL TIMES    6000        61  TO   69         7
C        7        TURNAROUND      7000        71  TO   79         6
C        8        WING SERVICE    8000        81  TO   89         4
C        9        MMT SERVICE     9000        91  TO   99         6
C       10        SQDN SERVICE   10000       101  TO  109         4
C       11        MAINTENANCE    11000       111  TO  119         3
C       12        STATISTICS     12000       121  TO  129         5
C       13        MTBF DIST.     13000       131  TO  139         7
C
C
C     EACH SECTION HAS ITS OWN COMMON BLOCK (IF REQ'D) AND DATA STATE-
C     MENTS FOR USER PROVIDED INFORMATION.  THE COMMON BLOCKS FOLLOW:
C
C     USER PROVIDED COMMON BLOCKS.
C
C     SECTION 1, UTILITY COMMON BLOCK:
      COMMON/UCOM1/LORSH(24),NAFSH,LTOW(24),NTOW,LBAT(24),NBAT,
     &            FFRGUN2,FFRGUN3,PGUNEXP,PGUNRUN,PEMDROP,PHUNGBM,
     &            PATTR2(3),PATTR3(3),PDAM2(3),PDAM3(3),PDL(5),
     &            DURMIN(3),DURMOD(3),DURMAX(3),PRLGEN(3),PRLWING(3)
C
C     SECTION 2, PARKING COMMON BLOCK:
      COMMON/UCOM2/NPARK(6,50,2),NTYPE(6,3)
C
C     SECTION 5, ALTER CODE COMMON BLOCK:
      COMMON/UCOM5/SYS(6,4),SYSTOL(6),MTBF(6),ALP(6),BET(6),NBATREP(5),
     &            ERUNMIN,ERUNMAX
C
C     SECTION 6, TRAVEL TIME COMMON BLOCK:
      COMMON/UCOM6/DIST(13,13),TOW(3),TAXI(3),CREW(3)
C
C     SECTION 7, TURNAROUND COMMON BLOCK:
```

```
      COMMON/UCOM7/NOCENT(3),NOWING(3),CPCENT,CPWING,CPINTR,
     &             CDRATE,CNDMAX,PERLEFT(3),REFRAT(3),UPTIME,DNTIME,
     &             DARMNOR(3),DARMHUG(3),DARMRUN(3),DARMEXP(3),
     &             RARMNOR(3),RARMHUG(3),RARMGUN(3),RARMBGN(3)
C
C     SECTION 8, WING SERVICE
      COMMON/UCOM8/MIN1(5),MODE1(5),MAX1(5),MIN2(5),MODE2(5),MAX2(5),
     &             MIN3(5),MODE3(5),MAX3(5),MIN4(5),MODE4(5),MAX4(5),
     &             MIN5(5),MODE5(5),MAX5(5),MIN6(5),MODE6(5),MAX6(5),
     &             WSH1INT(3),WSH3INT(3)
C
C     SECTION 9, MMT SERVICE
      COMMON/UCOM9/MMTMIN(6,2),MMTMOD(6,2),MMTMAX(6,2)
C
C     SECTION 10, SQUADRON SERVICE
      COMMON/UCOM10/MINI(5),MODEI(5),MAXI(5)
C
C     THE FOLLOWING ARE VARIABLE TYPE DECLARATIONS
C
      REAL MIN1,MODE1,MAX1,MIN2,MODE2,MAX2,MIN3,MODE3,MAX3,
     &     MIN4,MODE4,MAX4,MIN5,MODE5,MAX5,MIN6,MODE6,MAX6,
     &     MINI,MODEI,MAXI,MMTMIN,MMTMOD,MMTMAX,MTBF
      INTEGER MXCODE(7),LEV(6)
      LOGICAL NNTOW, NNCRSH, NNBAT
C
C
C
C
C
      IF(LEVPRT.GE.6.AND.TNOW.GE.BPRT.AND.TNOW.LE.EPRT)THEN
        PRINT*,'FUNCTION ',IFN,' CALLED, TIME ',TNOW,', A/C ',ATRIB(2)
      ENDIF
C
C     BRANCH TO THE SECTION OF THE USER FUNCTION DESIRED.
      I = IFN/10
      GOTO(1000,2000,3000,4000,5000,6000,7000,8000,9000,10000,
     +     11000,12000,13000),I
C
C
C
C
C
C
C
```

```
C     *********
C     SECTION 1, UTILITY -- USERF(11) TO USERF(19)
C     *********
 1000 CONTINUE
C
C     GLOSSARY OF VARIABLES USED IN THIS SECTION
C     COMMON/UCOM1/LCRSH(24),NCRSH,LTOW(24),NTOW,LBAT(24),NBAT,
C    &            PFRGUN2,PFRGUN3,PGUNEXP,PGUNRUN,PBMDROP,PHUNGBM,
C    &            PATTR2(3),PATTR3(3),PDAM2(3),PDAM3(3),PDL(5),
C    &            DURMIN(3),DURMOD(3),DURMAX(3),PRLCENT(3),PRLWING(3)
C     INTEGER MXCODE(7)
C     LOGICAL,NNTOW,NNCRSH,NNBAT
C
C     INTEGER VALUE VARIABLES:
C        LCRSH - ENCODE LEVELS WHICH CAUSE THE A/C TO CRASH.
C        NCRSH - NUMBER OF ENCODED LEVELS IN LCRSH.
C        LTOW  - ENCODED MX LEVELS WHICH CAUSE AN A/C TO BE TOWED.
C        NTOW  - NUMBER OF ENCODED LEVELS IN LTOW.
C        LBAT  - ENCODED MX & BATTLE DAMAGE LEVELS THAT CAUSE THE A/C
C                TO CRASH.
C        NBAT  - NUMBER OF ENCODED LEVELS IN LBAT.
C        NFUNC - FUNCTION BEING ACCESSED,
C        MXCODE- TEMPORARY STORAGE TO DECODE THE MX FAILURE CODE.
C        NDIV  - TEMPORARY DIVISOR, USED TO DECODE VALUES.
C        NCODE - TEMPORARY VALUE OF NUMBER BEING DECODED.
C        NVAL  - INDIVIDUAL DIGIT OF ENCODED NUMBER.
C
C
C     LOGICAL VALUE VARIABLES:
C        NNTOW - FLAG USED IN DETERMINING IF A/C REQUIRES TOWING.
C        NNCRSH- FLAG USED IN DETERMINING IF A/C CRASHED IN FLIGHT.
C        NNBAT - FLAG USED IN DETERMINING IF A/C CRASHED AFTER OR
C                DURING A MISSION.
C
C     END OF GLOSSARY.
C
C
C     DETERMINE WHICH USER FUNCTION IS BEING ACCESSED.
      NFUNC = MOD(IFN,10)
      GO TO(1100,1200,1300,1400,1500,1600,1700,1800,1900),NFUNC
C
C
C
C     FUNCTION USERF(11)
C     THIS ROUTINE ASSIGNS THE CORRECT STATUS TO EACH PILOT BEING
C     ENTERED INTO THE SYSTEM. GLOBAL VARIABLES XX(61),XX(62),XX(63),
C     AND XX(71) ARE USED TO GENERATE THE STATUS, THEY ARE:
C        XX(61) - NUMBER OF PILOTS ON QRA ALERT
C        XX(62) - TOTAL NUMBER OF PILOTS QRA QUALIFIED (INCLUDING
C                 THOSE ON ALERT).
```

```
C       XX(63) - TOTAL NUMBER FLIGHT LEAD QUALIFIED PILOTS (INCLUDING
C              ALL PILOTS THAT ARE QRA QUALIFIED), AND
C       XX(71) - COUNTER OF PILOTS BEING GENERATED.
C       XX(50) - EQUALS 1 IF RESUPPLY AIRCRAFT.
C       XX(57) - NUMBER OF ORIGINAL A/C PER SQUADRON.
C       XX(58) - NUMBER OF ORIGINAL PILOTS PER SQUADRON.
C
 1100 IF(TNOW.LT.0.1)THEN
C       ASSIGN STATUS OF PILOTS INITIALLY ON STATION
        USERF = 0.0
        IF(XX(71).LE.XX(63)) USERF = 1.0
        IF(XX(71).LE.XX(62)) USERF = 2.0
        IF(XX(71).LE.XX(61)) USERF = 3.0
      ELSE
C       ASSIGN STATUS OF RESUPPLY PILOTS
C        ASSUME SAME RATIO OF PILOTS FOR EACH
C        CATEGORY AS ORIGINALLY ON STATION
        USERF=0.0
        IF(XX(71).LE.XX(63)*XX(57)/XX(58))USERF=1.0
        IF(XX(71).LE.XX(62)*XX(57)/XX(58))USERF=2.0
      ENDIF
      RETURN
C
C
C     FUNCTION USERF(12)
C     THIS ROUTINE DETERMINES IF TOWING IS REQUIRED; IF SO, RETURN A 1.
C     IT DOES THIS BY SEEING IF EACH DIGIT OF THE MX FAILURE CODE IS
C     GREATER THAN OR EQUAL TO THE RESPECTIVE DIGIT IN LTOW BEING TESTED
C     IF THIS CONDITION IS MET FOR ANY ENCODED NUMBER OF LTOW, THEN THE
C     A/C MUST BE TOWED AND USERF EQUALS 1.
C
C
 1200 NDIV = 1000000
      NCODE = ATRIB(18)
C
C     DECODE THE MX FAILURE CODE FOR TAXING
      DO 1210 I = 1,6
        NCODE = MOD(NCODE,NDIV)
        NDIV = NDIV/10
        MXCODE(I) = NCODE/NDIV
 1210 CONTINUE
C
      J = 1
 1220 CONTINUE
        NNTOW = .TRUE.
        NDIV  = 1000000
        NCODE = LTOW(J)
        DO 1230 I = 1,6
C         DECODE THE TOWING CODE AND COMPARE TO MX FAILURE CODE
          NCODE = MOD(NCODE,NDIV)
```

```
  998                     NDIV  = NDIV/10
  999                    NVAL  = NCODE/NDIV
 1000           C        IF MX FAILURE LEVEL < USER PROVIDED CODE FOR
 1001           C         ANY GIVEN DIGIT, THEN DO NOT TOW.
 1002                    IF(MXCODE(I).LT.NVAL) NNTOW = .FALSE.
 1003            1230    CONTINUE
 1004                    J = J + 1
 1005           C     IF NO MATCH AND MORE ENCODED NUMBERS, TRY AGAIN
 1006                 IF(J.LE.NTOW.AND..NOT.NNTOW) GO TO 1220
 1007                 USERF = 0.0
 1008                 IF(NNTOW) USERF = 1.0
 1009                 RETURN
 1010           C
 1011           C
 1012           C     FUNCTION USERF(13)
 1013           C     THIS ROUTINE DETERMINES IF AN A/C CRASHED DUE TO A MX FAILURE
 1014           C     WHILE IN THE AIR.  IT OPERATES SIMILAR TO USERF(12).
 1015           C
 1016            1300 NDIV  = 1000000
 1017                 NCODE = ATRIB(18)
 1018           C
 1019                 DO 1310 I = 1,6
 1020                    NCODE = MOD(NCODE,NDIV)
 1021                    NDIV  = NDIV/10
 1022                    MXCODE(I) = NCODE/NDIV
 1023            1310 CONTINUE
 1024           C
 1025                 J = 1
 1026            1320 CONTINUE
 1027                    NNCRSH = .TRUE.
 1028                    NDIV   = 1000000
 1029                    NCODE  = LCRSH(J)
 1030           C
 1031                    DO 1330 I = 1,6
 1032                       NCODE = MOD(NCODE,NDIV)
 1033                       NDIV  = NDIV/10
 1034                       NVAL  = NCODE/NDIV
 1035                       IF(MXCODE(I).LT.NVAL) NNCRSH = .FALSE.
 1036            1330    CONTINUE
 1037                    J = J + 1
 1038                 IF(J.LE.NCRSH.AND..NOT.NNCRSH) GO TO 1320
 1039           C
 1040                 USERF = 0.0
 1041                 IF(NNCRSH) USERF = 1.0
 1042                 RETURN
 1043           C
 1044           C
 1045           C     FUNCTION USERF(14)
 1046           C     THIS ROUTINE DETERMINES IF AN A/C CRASHED DURING/FOLLOWING A
 1047           C     MISSION DUE TO A COMBINATION OF MX FAILURES AND BATTLE DAMAGE
```

```
1048            C     FAILURES.  IT IS SIMILAR TO USERF(12), EXCEPT THE BATTLE DAMAGE
1049            C     IS ALSO CONSIDERED.
1050            C
1051             1400 NDIV  = 1000000
1052                  NCODE = ATRIB(18)
1053            C
1054                  MXCODE(7) = ATRIB(16)
1055                  DO 1410 I = 1,6
1056                     NCODE = MOD(NCODE,NDIV)
1057                     NDIV  = NDIV/10
1058                     MXCODE(I) = NCODE/NDIV
1059             1410 CONTINUE
1060            C
1061                  J = 1
1062             1420 CONTINUE
1063                     NNBAT = .TRUE.
1064                     NDIV  = 10000000
1065                     NCODE = LBAT(J)
1066            C
1067                     DO 1430 I = 1,7
1068                        NCODE = MOD(NCODE,NDIV)
1069                        NDIV  = NDIV/10
1070                        NVAL  = NCODE/NDIV
1071                        IF(MXCODE(I).LT.NVAL) NNBAT = .FALSE.
1072              1430    CONTINUE
1073            C
1074                     J = J + 1
1075                  IF(J.LE.NBAT.AND..NOT.NNBAT) GO TO 1420
1076            C
1077                  USERF = 0.0
1078                  IF(NNBAT) USERF = 1.0
1079                  RETURN
1080            C
1081            C
1082            C     FUNCTION USERF(15)
1083            C     THIS ROUTINE DETERMINES WHAT OCCURS TO THE A/C DURING THE MISSION.
1084            C     IT RETURNS THE MISSION DURATION THROUGH THE NAME.  IT SETS THE
1085            C     BATTLE DAMAGE CODE, ATRIB(16), TO A LEVEL OF 0 TO 5, OR 99 IF KILL-
1086            C     ED. THE DURATION AND BATTLE DAMAGE CODE ARE BASED ON THE AREA THE A/C
1087            C     IS GOING TO, WHICH IS A FUNCTION OF TANK CONFIGURATION, ATRIB(12)
1088            C     AND THE NUMBER OF A/C IN THE FLIGHT.  IT ALSO DETERMINES THE WEA-
1089            C     PON STATUS: ATRIB(9),ATRIB(10), AND ATRIB(11); AND ANY CHANGES IN
1090            C     TANK CONFIGURATION.  ADDITIONALLY, IT INSURES THAT EACH A/C ON
1091            C     THE SAME MISSION RECEIVES THE SAME SORTIE DURATION.
1092            C
1093            C     MISSION NUMBER AND NUMBER OF A/C IN THE FLIGHT.
1094             1500 MSN = ATRIB(46)
1095                  NAC = ATRIB(44)
1096            C
1097            C     DETERMINE MISSION DURATION AND TO WHICH AREA THE A/C IS GOING
```

```
 1098            C
 1099                 IF(XX(MSN).EQ.0.0) THEN
 1100                    XX(97) = ATRIB(47)
 1101                    IF(XX(97).EQ.1.0) THEN
 1102                       MSNFLW(1) = MSNFLW(1) + 1
 1103                    ELSEIF(XX(97).EQ.2.0) THEN
 1104                       IF(MSNFLW(2).GE.MSNRQA2) THEN
 1105                          XX(97) = 1.0
 1106                          MSNFLW(1) = MSNFLW(1) + 1
 1107                       ELSE
 1108                          MSNFLW(2) = MSNFLW(2) + 1
 1109                       ENDIF
 1110                    ELSE
 1111                       MSNFLW(3) = MSNFLW(3) + 1
 1112                    ENDIF
 1113                    NAR = XX(97)
 1114                    XX(MSN) = TRIAG(DURMIN(NAR),DURMOD(NAR),DURMAX(NAR),3)
 1115                 ENDIF
 1116                 ATRIB(47) = XX(97)
 1117            C
 1118                 USERF = XX(MSN)
 1119            C
 1120            C    PRINT STATEMENTS
 1121            C
 1122                 IF(LEVPRT.GE.3.AND.TNOW.GE.BPRT.AND.TNOW.LE.EPRT)THEN
 1123                    PRINT*,'-AIRCRAFT NUMBER ',ATRIB(2),' IS ASSIGNED TO MISSION ',
 1124                &          MSN,', CURRENT TIME IS ',TNOW
 1125                 ENDIF
 1126                 IF(LEVPRT.GE.4.AND.TNOW.GE.BPRT.AND.TNOW.LE.EPRT)THEN
 1127                    PRINT*,' MISSION DURATION IS ',XX(MSN)
 1128                    PRINT*,' AREA IS ',ATRIB(47),' AND # TANKS IS ',ATRIB(12)
 1129                    PRINT*,' TOTAL SORTIES GENERATED SO FAR IS ',XX(94)
 1130                 ENDIF
 113.            C
 1132            C
 1133            C    DETERMINE STATUS OF GUN.
 1134                 X = DRAND(3)
 1135                 IF(NAC.EQ.2.AND.X.LE.PFRGUN2) ATRIB(9) = 0.0
 1136                 IF(NAC.EQ.3.AND.X.LE.PFRGUN3) ATRIB(9) = 0.0
 1137            C    IF THE GUN HAS BEEN FIRED, DETERMINE IF IT MALFUNCTIONED.
 1138                 IF(ATRIB(9).EQ.0.0) THEN
 1139                    X = DRAND(3)
 1140                    IF(X.LE.PGUNEXP+PGUNRUN) ATRIB(9) = 2.0
 1141                    IF(X.LE.PGUNEXP) ATRIB(9) = 3.0
 1142                 ENDIF
 1143            C
 1144            C    DETERMINE STATUS OF BOMBS.
 1145                 X = DRAND(3)
 1146                 IF(X.LE.PBMDROP) ATRIB(10) = 0.0
 1147            C    IF BOMBS ARE RELEASED, DETERMINE IF BOMBS HANG UP.
```

```
1148                IF(ATRIB(10).EQ.0.0) THEN
1149                  X = DRAND(3)
1150                  IF(X.LE.PHUNGBM) ATRIB(10) = 2.0
1151                ENDIF
1152          C
1153          C     DETERMINE MISSILE STATUS
1154                ATRIB(11) = ATRIB(11)
1155          C
1156          C     DETERMINE STATUS OF EXTERNAL TANKS
1157                NA = ATRIB(12)
1158                NW = NOWING(NA)
1159                NC = NOCENT(NA)
1160                NL = ATRIB(47)
1161          C
1162                X = DRAND(3)
1163                IF(X.LE.PRLWING(NL)) NW = 0
1164                IF(X.LE.PRLCENT(NL)) NC = 0
1165          C     KNOWING THE NUMBER OF WING TANKS (NW) AND THE NUMBER OF
1166          C     CENTERLINE TANKS (NC), DETERMINE THE A/C TANK CONFIGURATION
1167          C     CODE AND SET ATRIB(12).
1168                IF(NW.EQ.0.AND.NC.EQ.0) THEN
1169                   ATRIB(12) = 0.0
1170                ELSE
1171                   DO 1520 I = 1,NA
1172                      IF(NOCENT(I).EQ.NC.AND.NOWING(I).EQ.NW) ATRIB(12) = I
1173           1520    CONTINUE
1174                ENDIF
1175          C
1176          C     DETERMINE ATTRITION, (FUNCTION OF AREA AND # OF A/C IN FLIGHT)
1177          C
1178                X = DRAND(3)
1179                IF(NAC.EQ.2.AND.X.LE.PATTR2(NL)) ATRIB(16) = 99.0
1180                IF(NAC.EQ.3.AND.X.LE.PATTR3(NL)) ATRIB(16) = 99.0
1181          C
1182          C     DETERMINE BATTLE DAMAGE CODE, FUNCTION OF AREA AND NUMBER
1183          C     OF A/C IN THE FLIGHT.
1184          C
1185          C
1186                IF(ATRIB(16).NE.99.0)THEN
1187                ATRIB(16) = 0.0
1188                X = DRAND(3)
1189                IF(NAC.EQ.2.AND.X.LE.PDAM2(NL)) ATRIB(16) = 5.0
1190                IF(NAC.EQ.3.AND.X.LE.PDAM3(NL)) ATRIB(16) = 5.0
1191                IF(ATRIB(16).EQ.5.0) THEN
1192                   X = DRAND(3)
1193                   IF(X.LE.PDL(1)+PDL(2)+PDL(3)+PDL(4)) ATRIB(16) = 4.0
1194                   IF(X.LE.PDL(1)+PDL(2)+PDL(3)) ATRIB(16) = 3.0
1195                   IF(X.LE.PDL(1)+PDL(2)) ATRIB(16) = 2.0
1196                   IF(X.LE.PDL(1)) ATRIB(16) = 1.0
1197                ENDIF
```

```
 1198                ENDIF
 1199          C
 1200          C
 1201          C
 1202          C
 1203                RETURN
 1204          C
 1205          C
 1206          C
 1207          C
 1208          C     FUNCTION USERF(16)
 1209          C     INITIAL DISTRIBUTION OF FAILURE CODES FOR NON-OPERATIONAL A/C.
 1210          C
 1211           1600 NSUM = 0
 1212          C     A 50/50 CHANCE FOR EACH SYSTEM TO FAIL.  IF A SYSTEM IS TO
 1213          C      FAIL, THE LEVEL OF FAILURE IS UNIFORMLY DISTRIBUTED.
 1214                DO 1610 I = 1,6
 1215          C
 1216                   IF(DRAND(3).LT.0.50) THEN
 1217                      NSUM = NSUM*10 + IFIX(DRAND(3)*5.999)
 1218                   ELSE
 1219                      NSUM = NSUM*10
 1220                   ENDIF
 1221          C
 1222           1610 CONTINUE
 1223          C
 1224                USERF = NSUM
 1225                RETURN
 1226          C
 1227          C
 1228          C     FUNCTION USERF(17)
 1229          C     OBTAIN CURRENT SQUADRON NUMBER FOR THE A/C DESIGNATED
 1230          C       BY ATRIB(2)
 1231          C
 1232           1700 USERF = NCSQ(IFIX(ATRIB(2)))
 1233                RETURN
 1234          C
 1235          C
 1236          C     FUNCTION USERF(18)
 1237          C     SET NCSQ TO CURRENT SQ# FOR GIVEN TAIL #
 1238          C
 1239           1800 USERF = XX(72)
 1240                NCSQ(IFIX(XX(72))) = ATRIB(1)
 1241                RETURN
 1242          C
 1243          C
 1244          C     FUNCTION USERF(19)
 1245          C     MARK THE A/C WITH TAIL #, ATRIB(2), DESTROYED.
 1246           1900 USERF = ATRIB(2)
 1247                NCSQ(IFIX(ATRIB(2))) = 999999
```

```
 1248                RETURN
1249          C
1250          C
1251          C     END OF SECTION 1, UTILITY.
1252          C
```

```
1253            C     *********
1254            C     SECTION 2, PARKING --  USERF(21) TO USERF 23)
1255            C     *********
1256             2000 CONTINUE
1257            C
1258            C     GLOSSARY OF VARIABLES USED IN THIS SECTION.
1259            C     COMMON/UCOM2/NPARKING(6,50,2),NTYPE(6,3)
1260            C
1261            C     INTEGER vALUE VARIABLES:
1262            C        NPARK - PARKING SPACES FOR THE SIX SQDN'S:
1263            C                   INDEX 1, SQUADRON NUMBER
1264            C                   INDEX 2, PARKING SPOT (SHELTERS,REVETMENT,DISPERSED)
1265            C                   INDEX 3, TWO SPOTS (IN SHELTERS) FOR NIGHT.
1266            C        NTYPE - THE NUMBER OF EACH TYPE OF PARKING FOR THE SIX SQDN'S
1267            C                   INDEX 1, SQUADRON NUMBER
1268            C                   INDEX 2, TYPE OF PARKING SPOT - NTYPE(#,X) WHERE
1269            C                            X=1, NUMBER OF SHELTERS FOR SQDN #,
1270            C                            X=2, NUMBER OF REVETMENTS FOR SQDN #,
1271            C                            X=3, NUMBER OF DISPERSED SPOTS FOR SQDN #.
1272            C        NSQDN - SQUADRON NUMBER (1 TO 6).
1273            C        NTAIL - AIRCRAFT TAIL NUMBER.
1274            C        NCNT  - INDEX VARIABLE.
1275            C        JCNT  - INDEX VARIABLE.
1276            C        NFUNC - FUNCTION BEING ACCESSED.
1277            C     END OF GLOSSARY
1278            C
1279            C
1280            C     DETERMINE WHICH USER FUNCTION IS BEING ACCESSED IN THIS SECTION
1281                  NFUNC = MOD(IFN,10)
1282                  GO TO (2100,2200,2300),NFUNC
1283            C
1284            C
1285            C     FUNCTION USERF(21)
1286            C     THIS ROUTINE PARKS AIRCRAFT DURING THE NORMAL DAY TIME OPERATION
1287            C     OF THE AIRFIELD.  IT USES THE RULE OF TRYING TO PARK AN AIRCRAFT
1288            C     IN A SHELTER FIRST, IF ONE NOT AVAILABLE, THEN A REVETMENT, ELSE
1289            C     IF NOTHING BETTER IS OPEN, DISPERSED IN THE OPEN.  IF THE TOTAL
1290            C     NUMBER OF PARKING SPOTS IS EXCEEDED, A WARNING MESSAGE IS PRINTED.
1291            C
1292             2100 NCNT  = 1
1293                  NSQDN = ATRIB(1)
1294                  NTAIL = ATRIB(2)
1295            C
1296             2110 IF(NPARK(NSQDN,NCNT,1).EQ.0) GO TO 2120
1297                  NCNT = NCNT + 1
1298                  IF(NCNT.LE.50) GO TO 2110
1299                  USERF = 99.0
1300                  PRINT*,'  AIRCRAFT ',NTAIL,' HAS NO PARKING SPOT.'
1301                  RETURN
1302            C
```

```
1303            C     ONCE A SPOT IS FOUND, THE TYPE CODE MUST BY SET:
1304           C        1=SHELTER, 2=REVETMENT, 3=DISPERSED.
1305            2120 USERF = 1.0
1306                 IF(NCNT.GT.NTYPE(NSQDN,1)) USERF = 2.0
1307                 IF(NCNT.GT.NTYPE(NSQDN,2)+NTYPE(NSQDN,1)) USERF = 3.0
1308                 NPARK(NSQDN,NCNT,1)=NTAIL
1309                 RETURN
1310           C
1311           C
1312           C     FUNCTION USERF(22)
1313           C     THIS ROUTINE UNPARKS AN AIRCRAFT.  IT SETS THE PARKING SPOT TO 0
1314           C     AND RETURNS A VALUE OF 0.  IF AIRCRAFT IS NOT FOUND IN PARKING,
1315           C     A WARNING MESSAGE IS PRINTED.
1316           C
1317            2200 NCNT  = 1
1318                 JCNT = 1
1319                 NSQDN = ATRIB(1)
1320                 NTAIL = ATRIB(2)
1321           C
1322            2210 IF(NPARK(NSQDN,NCNT,JCNT).EQ.NTAIL) GO TO 2220
1323                 NCNT = NCNT + 1
1324                 IF(NCNT.LE.50) GO TO 2210
1325           C
1326                 NCNT = 1
1327                 JCNT = JCNT + 1
1328                 IF(JCNT.LE.2) GO TO 2210
1329           C
1330                 USERF = 0.0
1331                 PRINT*,'  AIRCRAFT ',NTAIL,' CAN NOT BE LOCATED IN PARKING.'
1332                 RETURN
1333           C
1334            2220 NPARK(NSQDN,NCNT,JCNT) = 0
1335                 USERF = 0.0
1336                 RETURN
1337           C
1338           C
1339           C     FUNCTION USERF(23)
1340           C     THIS ROUTINE PARKS THE AIRCRAFT THAT ARE INITIALLY BROKEN.  IT
1341           C     PARKS THEM IN REVERSE PRIORITY -- DISPERSED, REVETTED,SHELTERED.
1342           C     IF NO SPACE IS FOUND, A WARNING MESSAGE IS PRINTED.
1343           C
1344            2300 NCNT  = 50
1345                 NSQDN = ATRIB(1)
1346                 NTAIL = ATRIB(2)
1347           C
1348            2310 IF(NPARK(NSQDN,NCNT,1).EQ.0) GO TO 2320
1349                 NCNT = NCNT - 1
1350                 IF(NCNT.NE.0) GO TO 2310
1351                 USERF = 99.0
1352                 PRINT*,'  NO PARKING SPACE AVAILABLE FOR AIRCRAFT ',NTAIL
```

```
 1353                RETURN
1354          C
1355          C     ONCE A SPOT IS FOUND, THE TYPE CODE MUST BE DETERMINED:
1356          C        1=SHELTER, 2=REVETMENT, 3=DISPERSED.
1357           2320 USERF = 1.0
1358                IF(NCNT.GT.NTYPE(NSQDN,1)) USERF = 2.0
1359                IF(NCNT.GT.NTYPE(NSQDN,2)) USERF = 3.0
1360                NPARK(NSQDN,NCNT,1) = NTAIL
1361                RETURN
1362          C
1363          C
1364          C     END OF SECTION 2, PARKING.
1365          C
1366          C
1367          C
```

```
C     **********
C     SECTION 3, EXAMINE CODE  --  USERF(31) TO USERF(39)
C     **********
 3000 CONTINUE
C
C     GLOSSARY OF VARIABLES USED IN THIS SECTION.
C
C     INTEGER VALUE VARIABLES:
C        MAX   - MAXIUM LEVEL FOUND.
C        NDIV  - VALUE USED IN DECODING MX FAILURE CODE.
C        NCODE - MX FAILURE CODE BEING EXAMINED.
C        NVAL  - TEMPORARY VALUE STORAGE LOCATION.
C        NFUNC - USER FUNCTION BEING SELECTED.
C     END OF GLOSSARY.
C
C
C     THE MX FAILURE CODE IS A SIX DIGIT INTEGER, EACH DIGIT REPRE-
C     SENTING THE LEVEL OF FAILURE  (0-5, 0 BEING NO PROBLEM TO
C     5 BEING A SERIOUS PROBLEM).  THE MOST SIGNIFICANT DIGIT OF
C     THE CODE IS FOR SYSTEM 1, THE LEAST SIGNIFICANT IS SYSTEM 6.
C     DETERMINE WHICH USER FUNCTION IS BEING ACCESSED.
      NFUNC = MOD(IFN,10)
      GOTO(3100,3200,3300,3400,3500,3600,3700,3800,3900),NFUNC
C
C
C     FUNCTION USERF(31)
C     OBTAIN LEVEL OF SYSTEM ONE  (DIGIT 1 OF MX FAILURE CODE).
 3100 NCODE = ATRIB(18)
      USERF = MOD(NCODE,1000000)/100000
      RETURN
C
C
C     FUNCTION USERF(32)
C     OBTAIN LEVEL OF SYSTEM TWO  (DIGIT 2 OF MX FAILURE CODE).
 3200 NCODE = ATRIB(18)
      USERF = MOD(NCODE,100000)/10000
      RETURN
C
C
C     FUNCTION USERF(33)
C     OBTAIN LEVEL OF SYSTEM THREE  (DIGIT 3 OF MX FAILURE CODE).
 3300 NCODE = ATRIB(18)
      USERF = MOD(NCODE,10000)/1000
      RETURN
C
C
C     FUNCTION USERF(34)
C     OBTAIN LEVEL OF SYSTEM FOUR (DIGIT 4 OF MX FAILURE CODE).
 3400 NCODE = ATRIB(18)
      USERF = MOD(NCODE,1000)/100
```

```
 1418                RETURN
 1419          C
 1420          C
 1421          C
 1422          C     FUNCTION USERF(35)
 1423          C     OBTAIN LEVEL OF SYSTEM FIVE (DIGIT 5 OF MX FAILURE CODE).
 1424           3500 NCODE = ATRIB(18)
 1425                USERF = MOD(NCODE,100)/10
 1426                RETURN
 1427          C
 1428          C
 1429          C     FUNCTION USERF(36)
 1430          C     OBTAIN LEVEL OF SYSTEM SIX  (DIGIT 6 OF MX FAILURE CODE).
 1431           3600 NCODE = ATRIB(18)
 1432                USERF = MOD(NCODE,10)
 1433                RETURN
 1434          C
 1435          C
 1436          C     FUNCTION USERF(37)
 1437          C     ROUTINE TO DETERMINE THE MAXIUM LEVEL OF ALL MX FAILURES.
 1438           3700 MAX      = 0
 1439                NDIV  = 1000000
 1440                NCODE = ATRIB(18)
 1441          C
 1442                DO 3720 I = 1,6
 1443                   NCODE = MOD(NCODE,NDIV)
 1444                   NDIV  = NDIV/10
 1445                   NVAL  = NCODE/NDIV
 1446                   IF(NVAL.GT.MAX)MAX = NVAL
 1447           3720 CONTINUE
 1448          C
 1449                USERF = MAX
 1450                RETURN
 1451          C
 1452          C
 1453          C     FUNCTION USERF(38)
 1454          C     DETERMINE A/C PRIORITY FOR MX (WC/MMT) - LEAST BROKE FIRST.
 1455          C     THE VALUES OF ALL SYSTEMS WITH 4 AND 5 LEVEL VALUES ARE ADDED,
 1456          C     LOWEST SUM HAS FIRST PRIORITY.
 1457          C
 1458           3800 NCODE = ATRIB(18)
 1459                NDIV  = 1000000
 1460                NSUM  = 0
 1461                DO 3820 I = 1,6
 1462                   NCODE = MOD(NCODE,NDIV)
 1463                   NDIV  = NDIV/10
 1464                   NVAL  = NCODE/NDIV
 1465                   IF(NVAL.GE.4) NSUM = NSUM + NVAL
 1466           3820 CONTINUE
 1467                USERF = NSUM
```

```
1468                RETURN
1469          C
1470          C
1471          C     FUNCTION USERF(39)
1472          C     DETERMINE A/C PRIORITY FOR SQ MX - LEAST BROKE FIRST.
1473          C     THE VALUES OF ALL SYSTEMS WITH FAILURE LEVELS GREATER THAN OR
1474          C     EQUAL TO 2 ARE ADDED, LOWEST SUM HAS FIRST PRIORITY.
1475          C
1476           3900 NCODE = ATRIB(18)
1477                NDIV  = 1000000
1478                NSUM  = 0
1479                DO 3920 I = 1,6
1480                   NCODE = MOD(NCODE,NDIV)
1481                   NDIV  = NDIV/10
1482                   NVAL  = NCODE/NDIV
1483                   IF(NVAL.GE.2) NSUM = NSUM + NVAL
1484           3920 CONTINUE
1485                USERF = NSUM
1486                RETURN
1487          C
1488          C
1489          C     END OF SECTION 3, EXAMINE CODE.
1490          C
1491          C
1492          C
1493          C
1494          C
```

```
1495            C     *********
1496            C     SECTION 4, RESET CODE  --  USERF(41) TO USERF(47)
1497            C     *********
1498             4000 CONTINUE
1499            C
1500            C     GLOSSARY OF VARIABLES USED IN THIS SECTION.
1501            C
1502            C     INTEGER VALUE VARIABLES
1503            C        NCODE - MX FAILURE CODE BEING EXAMINED.
1504            C        NVAL  - TEMPORARY VALUE STORAGE LOCATION.
1505            C        NFUNC - USER FUNCTION BEING SELECTED
1506            C        NSUM  - NEW FAILURE CODE AFTER RESETTING ALL LEVELS.
1507            C        NDIV  - VALUE USED TO DECODE MX FAILURE CODE.
1508            C     END OF GLOSSARY.
1509            C
1510            C     THE MX FAILURE CODE IS A SIX DIGIT INTEGER, EACH DIGIT REPRE-
1511            C     SENTING THE SERIOUSNESS OF ONE OF SIX SYSTEMS.  THE DIGIT
1512            C     CAN TAKE ON THE VALUE OF 0 TO 5 (0 - NO PROBLEM, 5 - MOST
1513            C     SERIOUS PROBLEM).  THE MOST SIGNIFICANT DIGIT IS FOR SYSTEM
1514            C     1, THE LEAST SIGNIFICANT DIGIT IS FOR SYSTEM 6.
1515            C
1516            C     DETERMINE WHICH USER FUNCTION IS BEING ACCESSED.
1517                  NFUNC = MOD(IFN,10)
1518                  GOTO(4100,4200,4300,4400,4500,4600,4700),NFUNC
1519            C
1520            C
1521            C     FUNCTION USERF(41)
1522            C     ZERO OUT LEVEL OF SYSTEM ONE IN MX FAILURE CODE, UNLESS IT IS ONE.
1523             4100 NCODE = ATRIB(18)
1524                  NVAL  = NCODE/100000
1525                  IF(NVAL.NE.1)NVAL = 0
1526                  USERF = NVAL*100000 + MOD(NCODE,100000)
1527                  RETURN
1528            C
1529            C
1530            C     FUNCTION USERF(42)
1531            C     ZERO OUT LEVEL OF SYSTEM 2 IN MX FAILURE CODE, UNLESS IT IS ONE.
1532             4200 NCODE = ATRIB(18)
1533                  NVAL  = MOD(NCODE,100000)/10000
1534                  IF(NVAL.NE.1)NVAL = 0
1535                  USERF = (NCODE/100000*10+NVAL)*10000 + MOD(NCODE,10000)
1536                  RETURN
1537            C
1538            C
1539            C     FUNCTION USERF(43)
1540            C     ZERO OUT LEVEL OF SYSTEM 3 IN MX FAILURE CODE, UNLESS IT IS ONE.
1541             4300 NCODE = ATRIB(18)
1542                  NVAL  = MOD(NCODE,10000)/1000
1543                  IF(NVAL.NE.1)NVAL = 0
1544                  USERF = (NCODE/10000*10+NVAL)*1000 + MOD(NCODE,1000)
```

```
 1545                RETURN
1546            C
1547            C
1548            C     FUNCTION USERF(44)
1549            C     ZERO OUT LEVEL OF SYSTEM 4 IN MX FAILURE CODE, UNLESS IT IS ONE.
1550             4400 NCODE = ATRIB(18)
1551                  NVAL  = MOD(NCODE,1000)/100
1552                  IF(NVAL.NE.1)NVAL = 0
1553                  USERF = (NCODE/1000*10+NVAL)*100 + MOD(NCODE,100)
1554                  RETURN
1555            C
1556            C
1557            C     FUNCTION USERF(45)
1558            C     ZERO OUT LEVEL OF SYSTEM 5 IN MX FAILURE CODE, UNLESS IT IS ONE.
1559             4500 NCODE = ATRIB(18)
1560                  NVAL  = MOD(NCODE,100)/10
1561                  IF(NVAL.NE.1)NVAL = 0
1562                  USERF = (NCODE/100*10+NVAL)*10 + MOD(NCODE,10)
1563                  RETURN
1564            C
1565            C
1566            C     FUNCTION USERF(46)
1567            C     ZERO OUT LEVEL OF SYSTEM 6 IN MX FAILURE CODE, UNLESS IT IS ONE.
1568             4600 NCODE = ATRIB(18)
1569                  NVAL  = MOD(NCODE,10)
1570                  IF(NVAL.NE.1)NVAL = 0
1571                  USERF = NCODE/10*10 + NVAL
1572                  RETURN
1573            C
1574            C     FUNCTION USERF(47)
1575            C     ZERO OUT LEVEL OF ALL SYSTEMS IN MX FAILURE CODE, UNLESS IT IS 1.
1576             4700 NCODE = ATRIB(18)
1577                  NDIV  = 1000000
1578                  NSUM  = 0
1579            C
1580                  DO 4720 I = 1,6
1581                     NCODE = MOD(NCODE,NDIV)
1582                     NDIV  = NDIV/10
1583                     NVAL  = NCODE/NDIV
1584                     IF(NVAL.NE.1)NVAL = 0
1585                     NSUM  = NSUM*10 + NVAL
1586             4720 CONTINUE
1587            C
1588                  USERF = NSUM
1589                  RETURN
1590            C
1591            C
1592            C     END OF SECTION 4, RESET CODES
1593            C
1594            C
```

```
1595              C
1596              C
```

```
1597            C     *********
1598            C     SECTION 5, ALTER CODE -- USERF(51) TO USERF(53)
1599            C     *********
1600             5000 CONTINUE
1601            C
1602            C     GLOSSARY OF VARIABLES USED IN THIS SECTION.
1603            C     COMMON/UCOM5/SYS(6,4),SYSTOL(6),MTBF(6),ALP(6),BET(6),NBATREP(5),
1604            C    &            ERUNMIN,ERUNMAX
1605            C
1606            C     REAL VALUE VARIABLES:
1607            C        SYS   - CUMULATIVE PROBABILITIES TO DETERMINE LEVEL OF MX
1608            C                FAILURE FOR SYSTEM 1 THRU 6
1609            C                   INDEX 1 - SYSTEM NUMBER
1610            C     NBATREP CODES ARE SET IN INTLC.  THEY INDICATE THE APPROXIMATE
1611            C     LEVEL OF DAMAGE EACH SYSTEM WOULD HAVE FOR A GIVEN LEVEL OF
1612            C     BATTLE DAMAGE.  WITH THESE VALUES, THE ROUTINE TAKES THE
1613            C     GREATER OF THE TWO NUMBERS (MX FAILURE LEVEL, NBATREP LEVEL)
1614            C     FOR EACH SYSTEM (1-6) AND CREATES A NEW MX FAILURE CODE.
1615            C                   INDEX 2 - LEVEL OF REPAIR 1 TO 4 (CUM. OF 5 IS 1.00)
1616            C        SYSTOL- TOLERANCE WHICH ALLOWS MX FAILURES TO OCCUR DURING
1617            C                      PILOT PREFLIGHT; IF 0, IGNORE THAT SYSTEM.
1618            C        TOPER - A/C TOTAL MINUTES OF OPERATION (ENGINE RUNNING).
1619            C        X     - TEMPORARY STORAGE OF RANDOM DRAW.
1620            C        MTBF  - MEAN TIME BETWEEN FAILURE.
1621            C
1622            C     INTEGER VALUE VARIABLES:
1623            C        NCODE - TEMPORARY VALUE OF MX FAILURE CODE.
1624            C        NDIV  - USED TO DECODE MX FAILURE CODE.
1625            C        NVAL  - SINGLE DIGIT OF MX CODE BEING TESTED.
1626            C        NEW   - NEW DIGIT (LEVEL OF FAILURE FOR SYSTEM I).
1627            C        NSUM  - NEW MX FAILURE CODE.
1628            C
1629            C     END OF GLOSSARY.
1630            C
1631            C     THE MX FAILURE CODE IS A SIX DIGIT INTEGER, EACH DIGIT REPRE-
1632            C     SENTING THE SERIOUSNESS OF ONE OF SIX SYSTEMS.  THE DIGIT
1633            C     CAN TAKE ON THE VALUE OF 0 TO 5 (0 - NO PROBLEM, 5 - MOST
1634            C     SERIOUS PROBLEM).  THE MOST SIGNIFICANT DIGIT IS FOR SYSTEM
1635            C     1, THE LEAST SIGNIFICANT DIGIT IS FOR SYSTEM 6.
1636            C     DETERMINE WHICH USER FUNCTION IS BEING ACCESSED IN THIS SECTION
1637                  NFUNC = MOD(IFN,10)
1638                  GO TO (5100,5200,5300),NFUNC
1639            C
1640            C
1641            C     FUNCTION USERF(51)
1642            C     THIS ROUTINE DETERMINES IF A SYSTEM FAILED BY COMPARING CURRENT
1643            C     A/C TOTAL OPERATING TIME TO NEXT TIME OF FAILURE (NTOF) FOR THAT
1644            C     SYSTEM.  IF A SYSTEM FAILS, THE LEVEL IS DETERMINED STOCHASTICALLY
1645            C     THE GREATER OF THE NEW AND OLD LEVELS FOR A PARTICULAR SYSTEM IS
1646            C     ALWAYS CHOSEN.
```

```
1647            C
1648             5100 NCODE = ATRIB(18)
1649                  TOPER = ATRIB(7)
1650                  NDIV  = 1000000
1651                  NSUM  = 0
1652                  NSML = 0
1653                  NBIG = 0
1654            C
1655                  DO 5110 I = 1,6
1656                     NCODE = MOD(NCODE,NDIV)
1657                     NDIV  = NDIV/10
1658                     NVAL  = NCODE/NDIV
1659            C        SYSTEMS ARE ONLY ALLOWED TO BREAK ONCE, THE LEVEL IS THEN SET
1660            C        UNTIL THE NTOF IS RESET (MOST OF THE TIME AFTER MX).
1661            C
1662                     IF(NVAL.LE.1)THEN
1663                        NSML = NSML + 1
1664                        IF(TOPER.GE.ATRIB(18+I)) THEN
1665            C           IF A/C ENGINE TOTAL OPERATING TIME, TOPER, IS GREATER
1666            C           THAN OR EQUAL TO THE NTOF FOR ANY SYSTEM, A FAILURE
1667            C           LEVEL IS SET.
1668                           X = DRAND(3)
1669                           NVAL = 5
1670                           IF(X.LE.SYS(I,4)) NVAL = 4
1671                           IF(X.LE.SYS(I,3)) NVAL = 3
1672                           IF(X.LE.SYS(I,2)) NVAL = 2
1673                           IF(X.LE.SYS(I,1)) NVAL = 1
1674            C              IF THE LEVEL IS 1, THE NTOF IS RESET TO
1675            C              ALLOW THAT SYSTEM TO BREAK AGAIN, SINCE
1676            C              LEVEL 1 PROBLEMS ARE NOT FIXED BY MX.
1677                           IF(NVAL.EQ.1)THEN
1678                              ATRIB(18+I)=ATRIB(7) +
1679                 &            (1+BET(I)/ALP(I))*MTBF(I)*BETA(ALP(I),BET(I),2)
1680                           ELSE
1681                              NBIG = NBIG + 1
1682                           ENDIF
1683                        ENDIF
1684                     ENDIF
1685            C
1686                     NSUM = NSUM*10 + NVAL
1687              5110 CONTINUE
1688            C
1689            C     COUNT THE NUMBER OF A/C THAT HAVE BROKEN SINCE THEY WERE LAST
1690            C     IN MAINTENANCE.
1691                  IF(NSML.EQ.6.AND.NBIG.GT.0)XX(100) = XX(100) + 1
1692            C
1693                  USERF = NSUM
1694                  RETURN
1695            C
1696            C
```

```
1697            C     FUNCTION USERF(52)
1698            C     THIS ROUTINE ALTERS THE NEXT TIME OF FAILURE (NTOF) TO CURRENT TOTAL
1699            C     OPERATING MINUTES IF THE DIFFERENCE BETWEEN THEM FOR SYSTEM I IS .LE.
1700            C     SYSTOL(I) MINUTES.  USED TO CAUSE A PREFLIGHT MX FAILURE.
1701            C
1702             5200 TOPER = ATRIB(7)
1703                  DO 5210 I = 1,6
1704                     IF(ATRIB(18+I) - TOPER.LE.SYSTOL(I)) ATRIB(18+I) = TOPER
1705             5210 CONTINUE
1706                  USERF = ATRIB(18)
1707                  RETURN
1708            C
1709            C
1710            C     FUNCTION USERF(53)
1711            C     THIS ROUTINE COMBINES THE MX FAILURE CODE WITH THE BATTLE DAMAGE
1712            C     CODE TO INSURE MX IS PERFORMED ON BATTLE DAMAGE.  ALSO DETERMINES
1713            C     IF AN A/C IS REPAIRABLE OR JUNK(999999).
1714            C
1715            C     NC IS THE BATTLE DAMAGE LEVEL WHICH IS SET IN MISSION ROUTINE,
1716            C     USERF(15).
1717             5300 NC = ATRIB(16)
1718            C
1719                  IF(NBATREP(NC).EQ.999999) THEN
1720            C        A/C IS BEYOND REPAIR
1721                     ATRIB(18) = 999999.0
1722                  ELSE
1723            C        MODIFY MX CODE TO INCORPORATE THE BATTLE DAMAGE CODE
1724                     NSUM = 0
1725                     NDIV = 1000000
1726                     NCODE= ATRIB(18)
1727                     NBTL = NBATREP(NC)
1728            C
1729            C
1730                     DO 5320 I = 1,6
1731                        NCODE = MOD(NCODE,NDIV)
1732                        NBTL  = MOD(NBTL,NDIV)
1733                        NDIV  = NDIV/10
1734                        NVAL  = NCODE/NDIV
1735                        IF(NBTL/NDIV.GT.NVAL) NVAL = NBTL/NDIV
1736                        NSUM = NSUM*10 + NVAL
1737             5320    CONTINUE
1738                     ATRIB(16) = NSUM
1739                  ENDIF
1740                  USERF = ATRIB(18)
1741            C
1742                  RETURN
1743            C
1744            C     END OF SECTION 5, ALTER CODE.
1745            C
1746            C
```

```
 1747              C
1748               C
```

```
C     *********
C     SECTION 6, TRAVEL TIME -- USERF(61) TO USERF(67)
C     *********
 6000 CONTINUE
C
C     GLOSSARY OF VARIABLES USED IN THIS SECTION
C     COMMON/DIST(13,13),TOW(3),TAXI(3),CREW(3)
C
C     REAL VALUE VARIABLES:
C        DIST  - THE DISTANCE FROM POINT I (INDEX 1) TO POINT J (INDEX 2)
C        TOW   - THE LOWER BOUND, MODE, AND UPPER BOUND TRAVEL RATE.
C        TAXI  - THE LOWER BOUND, MODE, AND UPPER BOUND TRAVEL RATE.
C        CREW  - THE LOWER BOUND, MODE, AND UPPER BOUND TRAVEL RATE.
C        DIST  - MATRIX OF DISTANCES BETWEEN POINTS ON THE FIELD.
C
C     INTEGER VALUE VARIABLES:
C        I     - POINT ON FIELD FROM
C        J     - POINT ON FIELD TO
C
C     END OF GLOSSARY
C
C
C     DETERMINE FUNCTION BEING ACCESSED.
      NFUNC = MOD(IFN,10)
      GO TO(6100,6200,6300,6400,6500,6600,6700),NFUNC
C     FUNCTION USERF(61)
C     ROUTINE TO DETERMINE TAXI TIME FROM POINT I, ATRIB(13) TO
C     SQUADRON J, ATRIB(1)
C
 6100 I = IFIX(ATRIB(13))
      J = IFIX(ATRIB(1))
      USERF = DIST(I,J) / TRIAG(TAXI(1),TAXI(2),TAXI(3),3)
      RETURN
C
C
C     FUNCTION USERF(62)
C     ROUTINE TO DETERMINE TAXI TIME FROM POINT I, ATRIB(13) TO
C     WING AREA, CODE 9.
C
 6200 I = IFIX(ATRIB(13))
      J = 9
      USERF = DIST(I,J) / TRIAG(TAXI(1),TAXI(2),TAXI(3),3)
      RETURN
C
C
C     FUNCTION USERF(63)
C     DETERMINE TAXI TIME FROM SQDN I, ATRIB(13) TO ARMING AREA, CODE 7
C
 6300 I = IFIX(ATRIB(13))
      J = 7
```

```
 1799                 USERF = DIST(I,J) / TRIAG(TAXI(1),TAXI(2),TAXI(3),3)
 1800                 RETURN
 1801           C
 1802           C
 1803           C     FUNCTION USERF(64)
 1804           C     DETERMINE TAXI TIME FROM HOTPIT TO POINT J, ATRIB(13).
 1805           C
 1806            6400 I = 11
 1807                 J = IFIX(ATRIB(13))
 1808                 USERF = DIST(I,J) / TRIAG(TAXI(1),TAXI(2),TAXI(3),3)
 1809                 RETURN
 1810           C
 1811           C
 1812           C     FUNCTION USERF(65)
 1813           C     DETERMINE TOW TIME FROM POINT I, ATRIB(13) TO SQDN J, ATRIB(1)
 1814           C
 1815            6500 I = IFIX(ATRIB(13))
 1816                 J = IFIX(ATRIB(1))
 1817                 USERF = DIST(I,J) / TRIAG(TOW(1),TOW(2),TOW(3),3)
 1818                 RETURN
 1819           C
 1820           C
 1821           C     FUNCTION USERF(66)
 1822           C     DETERMINE TOW TIME FROM POINT I,ATRIB(13), TO WING AREA, SIDE 9.
 1823           C
 1824            6600 I = IFIX(ATRIB(13))
 1825                 J = 9
 1826                 USERF = DIST(I,J) / TRIAG(TOW(1),TOW(2),TOW(3),3)
 1827                 RETURN
 1828           C
 1829           C
 1830           C     FUNCTION USERF(67)
 1831           C     CREW TRAVEL TIME FROM POINT I, ATRIB(13) TO SQDN J, ATRIB(1).
 1832           C
 1833            6700 I = IFIX(ATRIB(13))
 1834                 J = IFIX(ATRIB(1))
 1835                 USERF = DIST(I,J) / TRIAG(CREW(1),CREW(2),CREW(3),3)
 1836                 RETURN
 1837           C
 1838           C
 1839           C     END OF SECTION 6, TRAVEL TIME.
 1840           C
 1841           C
 1842           C
```

```
C     *********
C     SECTION 7, TURNAROUND -- USERF(71) TO USERF(78)
C     *********
 7000 CONTINUE
C
C     GLOSSARY OF VARIABLES USED IN THIS SECTION
      COMMON/UCOM7/NOCENT(3),NOWING(3),CPCENT,CPWING,CPINTR,
     &GDRATE,GNDMAX,PERLEFT(3),REFRAT(3),UPTIME,DNTIME,
     &DARMNOR(3),DARMHOG(3),DARMRUN(3),DARMEXP(3),
     &RARMNOR(3),RARMHOG(3),RARMGUN(3),RARMBOM(3)
      COMMON/STATS/MAXCONF(6),NACTYPE(6,3),NOSQ(300),MSNRQAZ,MSNFLW(3)
C
C     REAL VALUE VARIABLES:
C        GNDTM  - GROUND RUNNING TIME OF A/C SINCE LAST REFUELING.
C        AIRTM  - AIR RUNNING TIME OF A/C SINCE LAST REFUELING.
C        GNDRATE- RATE OF FUEL CONSUMPTION ON GROUND.
C        AIRRATE- RATE OF FUEL CONSUMPTION IN AIR.
C        POLUSED- POL BURNED SINCE LAST REFUELING.
C        CENTKCP- CAPACITY OF EXTERNAL CENTERLINE TANK(3).
C        WGTKCAP- CAPACITY OF EXTERNAL WING TANKS.
C        CAPINT - INTERNAL CAPACITY OF FUEL FOR A/C.
C        AMTREM - AMOUNT REMAINING.
C        TOTVOL - TOTAL POL CAPACITY OF A/C.
C        GNDMAX - MAX GND TIME WITHOUT REFUELING.
C        UPRATE - RATE OF UPLOADING EXTERNAL TANKS.
C        DNRATE - RATE OF DOWN-LOADING EXTERNAL TANKS.
C        NCONF  - CURRENT A/C CONFIGURATION
C        NSQ    - A/C SQUADRON NUMBER
C        MAXCONF- CONFIGURATION NUMBER FOR SQUADRON
C        NACTYPE- # OF A/C PER SQDN PER CONFIGURATION
C
C
C     END OF GLOSSARY
      NFUNC = MOD(IFN,10)
      GO TO (7100,7200,7300,7400,7500,7600,7700,7800),NFUNC
C
C     FUNCTION USERF(71)
C     DETERMINE THE FUEL REQ'D: ATRIB(14),GROUND OPERATING TIME;
C     ATRIB(15), AIR OPERATING TIME. IF ATRIB(15) EQUALS 0, THEN
C     NO AIR TIME OR HOT PIT REFUELED
C
C     A/C TANK CONFIGURATION (NC)
 7100 NC = ATRIB(12)
C
      IF(ATRIB(15).NE.0.0) THEN
C        THE A/C HAS AIR TIME ON IT SINCE THE LAST REFUELING.
         AMTLEFT= CPINTR*TRIAG(PERLEFT(1),PERLEFT(2),PERLEFT(3),3)
         AMTLEFT= AMTLEFT - ATRIB(14)*GDRATE
         IF(AMTLEFT.LT.0.0) AMTLEFT = 0.0
         IF(NC.GT.0) THEN
```

```
1893                     TOTVOL = CPINTR + NOCENT(NC)*CPCENT + NOWING(NC)*CPWING
1894                  ELSE
1895                     TOTVOL = CPINTR
1896                  ENDIF
1897                  AMTREQ = TOTVOL - AMTLEFT
1898            C
1899               ELSEIF(ATRIB(14).GE.GNDMAX) THEN
1900                  IF(NC.GT.0) THEN
1901                     TOTVOL = CPINTR + NOCENT(NC)*CPCENT + NOWING(NC)*CPWING
1902                  ELSE
1903                     TOTVOL = CPINTR
1904                  ENDIF
1905                  AMTUSED= ATRIB(14)*CDRATE
1906                  IF(AMTUSED.GT.TOTVOL) AMTUSED = TOTVOL
1907                  AMTREQ= TOTVOL - AMTUSED
1908            C
1909               ELSE
1910                  AMTREQ = 0.0
1911               ENDIF
1912            C
1913               AMTREQ = AMTREQ*0.000720463
1914               USERF = AMTREQ
1915            C
1916               RETURN
1917            C
1918            C
1919            C     FUNCTION USERF(72)
1920            C     RECALCULATE FUEL REQUIRED DUE TO TANK RECONFIGURATION.  ATRIB(8)
1921            C     HAS NEW CONFIGURATION, WHILE ATRIB(12) HAS OLD CONFIGURATION.
1922            C
1923             7200 NC = ATRIB(8)
1924            C
1925               CPNEW = CPINTR + NOCENT(NC)*CPCENT + NOWING(NC)*CPWING
1926            C
1927               NC = ATRIB(12)
1928            C
1929               IF(NC.GT.0) THEN
1930                  CPOLD = CPINTR + NOCENT(NC)*CPCENT + NOWING(NC)*CPWING
1931               ELSE
1932                  CPOLD = CPINTR
1933               ENDIF
1934            C
1935               IF(CPOLD.LT.CPNEW) THEN
1936                  USERF = ATRIB(15) + (CPNEW - CPOLD)*0.000720463
1937               ELSEIF(CPOLD.GT.CPNEW) THEN
1938                  AMTREQ = ATRIB(15) - (CPOLD - CPNEW)*0.000720463
1939                  IF(AMTREQ.LT.0.0) AMTREQ = 0.0
1940                  USERF = AMTREQ
1941               ELSE
1942                  USERF = ATRIB(15)
```

```
1943                    ENDIF
1944          C
1945                RETURN
1946          C
1947          C
1948          C     FUNCTION USERF(73)
1949          C     RECONFIGURATION REQUIREMENT
1950          C     NACTYPE(NSQ,1) - # OF A/C OF CONFIGURATION 1 IN SQUADRON NSQ,
1951          C     NACTYPE(NSQ,2) - # OF A/C OF CONFIGURATION 2 IN SQUADRON NSQ,
1952          C     NACTYPE(NSQ,3) - # OF A/C OF CONFIGURATION 3 IN SQUADRON NSQ.
1953          C
1954          C
1955           7300 NSQ = ATRIB(1)
1956          C
1957          C
1958          C     IF A/C IS REFUELED AND READY TO GO THEN THE A/C IS
1959          C     NOT RECONFIGURED UNLESS IT HAS LESS TANKS THAN ITS
1960          C     SQUADRON REQUIRES FOR MAXCONF(NSQ).
1961          C
1962          C
1963                IF(ATRIB(15).EQ.0.0.AND.ATRIB(14).LT.GNDMAX) THEN
1964                   IF(ATRIB(12).GE.MAXCONF(NSQ))THEN
1965                      USERF = ATRIB(12)
1966                      NC = ATRIB(12)
1967                   ELSE
1968                      USERF = MAXCONF(NSQ)
1969                      NC = MAXCONF(NSQ)
1970                   ENDIF
1971                   NACTYPE(NSQ,NC) = NACTYPE(NSQ,NC) + 1
1972          C     ELSE, THE ROUTINE CHECKS TO SEE IF ALL ODD BALL CONFIGURATIONS
1973          C     ARE MATCHED, THEN IT IS ALLOWED TO RECONFIGURE TO THE SPECIFIED
1974          C     CONFIGURATION, MAXCONF(NSQ).
1975                ELSEIF(MAXCONF(NSQ).EQ.1) THEN
1976          C        MAXIUM ALLOWED CONFIGURATION FOR THE SQUADRON IS 1, UNLESS
1977          C        THERE ARE CONFIGURATION 2 AND/OR 3 A/C ALREADY IN THE
1978          C        READY POOL OF SQUADRON NSQ.
1979          C
1980                   IF(NACTYPE(NSQ,3).EQ.2) THEN
1981                      USERF = 3.0
1982                      NACTYPE(NSQ,3) = NACTYPE(NSQ,3) + 1
1983                   ELSEIF(NACTYPE(NSQ,2).EQ.2) THEN
1984                      USERF = 2.0
1985                      NACTYPE(NSQ,2) = NACTYPE(NSQ,2) + 1
1986                   ELSEIF(NACTYPE(NSQ,3).EQ.1) THEN
1987                      USERF = 3.0
1988                      NACTYPE(NSQ,3) = NACTYPE(NSQ,3) + 1
1989                   ELSEIF(NACTYPE(NSQ,2).EQ.1) THEN
1990                      USERF = 2.0
1991                      NACTYPE(NSQ,2) = NACTYPE(NSQ,2) + 1
1992                   ELSEIF(ATRIB(12).EQ.1.OR.ATRIB(12).EQ.2) THEN
```

```
 1993                      USERF = ATRIB(12)
1994                      NACTYPE(NSQ,IFIX(ATRIB(12)))=NACTYPE(NSQ,IFIX(ATRIB(12)))+1
1995                   ELSE
1996                      USERF = 1.0
1997                      NACTYPE(NSQ,1) = NACTYPE(NSQ,1) + 1
1998                   ENDIF
1999             C
2000                ELSEIF(MAXCONF(NSQ).EQ.2) THEN
2001             C     MAX SPECIFIED CONFIGURATION FOR SQUADRON NSQ IS 2.
2002             C
2003                   IF(NACTYPE(NSQ,3).EQ.2) THEN
2004                      USERF = 3.0
2005                      NACTYPE(NSQ,3) = NACTYPE(NSQ,3) + 1
2006                   ELSEIF(NACTYPE(NSQ,1).EQ.2) THEN
2007                      USERF = 1.0
2008                      NACTYPE(NSQ,1) = NACTYPE(NSQ,1) + 1
2009                   ELSEIF(NACTYPE(NSQ,3).EQ.1) THEN
2010                      USERF = 3.0
2011                      NACTYPE(NSQ,3) = NACTYPE(NSQ,3) + 1
2012                   ELSEIF(NACTYPE(NSQ,1).EQ.1) THEN
2013                      USERF = 1.0
2014                      NACTYPE(NSQ,1) = NACTYPE(NSQ,1) + 1
2015                   ELSE
2016                      USERF = 2.0
2017                      NACTYPE(NSQ,2) = NACTYPE(NSQ,2) + 1
2018                   ENDIF
2019             C
2020                ELSE
2021             C     MAX SPECIFIED CONFIGURATION FOR SQUADRON NSQ IS 3.
2022             C
2023                   IF(NACTYPE(NSQ,2).EQ.2) THEN
2024                      USERF = 2.0
2025                      NACTYPE(NSQ,2) = NACTYPE(NSQ,2) + 1
2026                   ELSEIF(NACTYPE(NSQ,1).EQ.2) THEN
2027                      USERF = 1.0
2028                      NACTYPE(NSQ,1) = NACTYPE(NSQ,1) + 1
2029                   ELSEIF(NACTYPE(NSQ,2).EQ.1) THEN
2030                      USERF = 2.0
2031                      NACTYPE(NSQ,2) = NACTYPE(NSQ,2) + 1
2032                   ELSEIF(NACTYPE(NSQ,1).EQ.1) THEN
2033                      USERF = 1.0
2034                      NACTYPE(NSQ,1) = NACTYPE(NSQ,1) + 1
2035                   ELSE
2036                      USERF = 3.0
2037                      NACTYPE(NSQ,3) = NACTYPE(NSQ,3) + 1
2038                   ENDIF
2039                ENDIF
2040             C
2041                RETURN
2042             C
```

```
 2043            C
2044            C     FUNCTION USERF(74)
2045            C     DETERMINE LENGTH OF TIME REQUIRED TO ALTER CONFIGURATION.
2046             7400 NC = ATRIB(8)
2047            C     CALCULATE THE NUMBER OF TANKS THE A/C HAD.
2048                  NEWTKS = NOCENT(NC) + NOWING(NC)
2049            C
2050            C     CALCULATE THE NUMBER OF NEW TANKS DESIRED.
2051                  NC = ATRIB(12)
2052                  IF(NC.EQ.0) THEN
2053                     NODTKS = 0
2054                  ELSE
2055                     NODTKS = NOCENT(NC) + NOWING(NC)
2056                  ENDIF
2057            C
2058            C     DETERMINE THE TIME REQUIRED TO ALTER THE CONFIGURATION.
2059                  IF(NEWTKS.GT.NODTKS) THEN
2060                     USERF = (NEWTKS - NODTKS)*UPTIME
2061                  ELSE
2062                     USERF = (NODTKS - NEWTKS)*DNTIME
2063                  ENDIF
2064            C
2065                  RETURN
2066            C
2067            C
2068            C
2069            C
2070            C     FUNCTION USERF(75)
2071            C     ROUTINE TO DETERMINE DEARM SERVICE TIME (CONSIDERS WEAPON STATUS)
2072            C     THE TIME IS THE SUM OF THE GUN SERVICING TIME AND BOMB
2073            C     SERVICING TIME.
2074            C
2075             7500 IF(ATRIB(10).LE.1) THEN
2076                     XBOMB = TRIAG(DARMNOR(1),DARMNOR(2),DARMNOR(3),3)
2077                  ELSE
2078                     XBOMB = TRIAG(DARMHUG(1),DARMHUG(2),DARMHUG(3),3)
2079                  ENDIF
2080            C
2081                  IF(ATRIB(9).LE.1) THEN
2082                     XGUN  = 0.0
2083                  ELSEIF(ATRIB(9).EQ.2) THEN
2084                     XGUN = TRIAG(DARMRUN(1),DARMRUN(2),DARMRUN(3),3)
2085                  ELSE
2086                     XGUN = TRIAG(DARMEXP(1),DARMEXP(2),DARMEXP(3),3)
2087                  ENDIF
2088            C
2089                  USERF = XBOMB + XGUN
2090            C
2091                  RETURN
2092            C
```

```
2093            C
2094            C     FUNCTION USERF(76)
2095            C     DETERMINE TIME TO REARM A/C,VARIES ACCORDING TO WEAPON STATUS.
2096            C     THE TIME IS THE SUM OF BOMB LOADING TIME, GUN LOADING TIME,
2097            C     AND TIME TO FIX ANY PROBLEMS.  IF THE A/C DOES NOT REQUIRE
2098            C     ANY SERVICE FOR ONE OF THE ABOVE, THE TIME FOR THAT ELEMENT
2099            C     IS ZERO.
2100            C
2101             7600 IF(ATRIB(10).EQ.1) THEN
2102                     XBOMB = 0.0
2103                  ELSE
2104                     XBOMB = TRIAG(RARMNOR(1),RARMNOR(2),RARMNOR(3),3)
2105                  ENDIF
2106            C
2107                  IF(ATRIB(9).EQ.1) THEN
2108                     XGUN = 0.0
2109                  ELSE
2110                     XGUN = TRIAG(RARMGUN(1),RARMGUN(2),RARMGUN(3),3)
2111                  ENDIF
2112            C
2113                  IF(ATRIB(9).GE.2) THEN
2114                     XPROB = TRIAG(RARMBGN(1),RARMBGN(2),RARMBGN(3),3)
2115                  ELSEIF(ATRIB(10).EQ.2) THEN
2116                     XPROB = TRIAG(RARMHUG(1),RARMHUG(2),RARMHUG(3),3)
2117                  ELSE
2118                     XPROB = 0.0
2119                  ENDIF
2120            C
2121                  ATRIB(9) = 1.0
2122                  ATRIB(10)= 1.0
2123                  USERF = XBOMB + XGUN + XPROB
2124                  RETURN
2125            C
2126            C
2127            C     FUNCTION USERF(77)
2128            C     DETERMINE IF A/C ALLOWED TO HOT PIT
2129            C     FUNCTION OF A/C LOCATION, SQUADRON NUMBER, AND CONFIGURATION.
2130            C     AVAILABILITY OF HOTPIT RESOURCES IS ONLY CONSIDERED AFTER
2131            C     THE A/C GETS TO THE HOTPIT.
2132             7700 USERF =0.0
2133            C
2134                  NSQ = NCSQ(IFIX(ATRIB(2)))
2135                  IF(ATRIB(3).NE.1.AND.ATRIB(12).LE.MAXCONF(NSQ).AND.
2136                 +  NUMRES(40).GT.0.AND.
2137                 +  (ATRIB(1).LE.3.AND.ATRIB(13).EQ.7.OR.
2138                 +  ATRIB(1).GT.3.AND.ATRIB(13).EQ.10)) USERF = 1.0
2139            C
2140                  RETURN
2141            C
2142            C
```

```
 2143            C     FUNCTION USERF(78)
2144            C     DETERMINE TIME REQ'D TO REFUEL
2145            C
2146             7800 USERF =(ATRIB(15) * 1388.0)/TRIAC(REFRAT(1),REFRAT(2),REFRAT(3),3)
2147                  RETURN
2148            C
2149            C
2150            C     END OF SECTION 7, TURNAROUND.
2151            C
2152            C
2153            C
```

```
C     *********
C     SECTION 8, WING SERVICE      --  USERF(81) TO USERF(84)
C     *********
 8000 CONTINUE
C
C     WING SERVICE FIXES LEVEL 4 AND LEVEL 5 PROBLEMS.  IT
C     ASSUMES THAT THE LEVEL3 AND LEVEL 2 PROBLEMS ARE
C     FIXED CONCURENTLY WITHOUT ADDITIONAL TIME BEING REQ'D
C     OR ADDITIONAL RESOURCES.
C     GLOSSARY OF VARIABLES:
C        COMMON/UCOM8/MIN1(5),MODE1(5),MAX1(5),MIN2(5),MODE2(5),MAX2(5),
C                     MIN3(5),MODE3(5),MAX3(5),MIN4(5),MODE4(5),MAX4(5),
C                     MIN5(5),MODE5(5),MAX5(5),MIN6(5),MODE6(5),MAX6(5),
C                     WSH1INT(3),WSH3INT(3)
C
C
C     THE SERVICE TIMES ARE A FUNCTION OF WHETHER OR NOT TWO ACTIV-
C     TIES ARE GOING ON AT ONCE.  IF SO, THE TIME IS THE MAXIMUM
C     TIME OF THE TWO PLUS AN INTERFERENCE TIME BECAUSE OF THE CON-
C     CURRENT ACTIVITIES.  IF NOT, THE TIME IS THE LENGTH OF TIME
C     REQUIRED TO FIX THE ONE PROBLEM.
C
      NFUNC = MOD(IFN,10)
      GO TO (8100,8200,8300,8400),NFUNC
C
C
C
C     FUNCTION USERF(81)
C     SERVICE TIME OF SHOP ONE.
C
 8100 NCODE   = ATRIB(18)
      LEVSYS3 = MOD(NCODE,10000)/1000
      LEVSYS4 = MOD(NCODE,1000)/100
C
      IF(LEVSYS3.GE.4) THEN
        F3 = TRIAG(MIN3(LEVSYS3),MODE3(LEVSYS3),MAX3(LEVSYS3),3)
      ELSE
        F3 = 0.0
      ENDIF
C
      IF(LEVSYS4.GE.4) THEN
        F4 = TRIAG(MIN4(LEVSYS4),MODE4(LEVSYS4),MAX4(LEVSYS4),3)
      ELSE
        F4 = 0.0
      ENDIF
C
      IF(F3.GE.F4.AND.F4.NE.0.0)THEN
        USERF = F3 + TRIAG(WSH1INT(1),WSH1INT(2),WSH1INT(3),3)
       ELSEIF(F4.GE.F3.AND.F3.NE.0.0)THEN
        USERF = F4 + TRIAG(WSH1INT(1),WSH1INT(2),WSH1INT(3),3)
```

```
2204                ELSE
2205                  USERF = F3 + F4
2206                ENDIF
2207          C
2208                RETURN
2209          C     FUNCTION USERF(82)
2210          C     SERVICE TIME OF WING SHOP TWO.
2211          C
2212          C
2213           8200 NCODE   = ATRIB(18)
2214                LEVSYS2 = MOD(NCODE,100000)/10000
2215          C
2216                IF(LEVSYS2.GE.4) THEN
2217                   USERF = TRIAG(MIN2(LEVSYS2),MODE2(LEVSYS2),MAX2(LEVSYS2),3)
2218                ELSE
2219                   USERF = 0.0
2220                ENDIF
2221          C
2222                RETURN
2223          C
2224          C
2225          C
2226          C     FUNCTION USERF(83)
2227          C     SERVICE TIME OF WING SHOP THREE.
2228          C
2229           8300 NCODE   = ATRIB(18)
2230                LEVSYS1 = MOD(NCODE,1000000)/100000
2231                LEVSYS5 = MOD(NCODE,100)/10
2232          C
2233                IF(LEVSYS1.GE.4) THEN
2234                   F1 = TRIAG(MIN1(LEVSYS1),MODE1(LEVSYS1),MAX1(LEVSYS1),3)
2235                ELSE
2236                   F1 = 0.0
2237                ENDIF
2238          C
2239                IF(LEVSYS5.GE.4) THEN
2240                   F5 = TRIAG(MIN5(LEVSYS5),MODE5(LEVSYS5),MAX5(LEVSYS5),3)
2241                ELSE
2242                   F5 = 0.0
2243                ENDIF
2244          C
2245                IF(F1.GE.F5.AND.F5.GT.0.0)THEN
2246                   USERF = F1 + TRIAG(WSH3INT(1),WSH3INT(2),WSH3INT(3),3)
2247                ELSEIF(F5.GE.F1.AND.F1.GT.0.0)THEN
2248                   USERF = F5 + TRIAG(WSH3INT(1),WSH3INT(2),WSH3INT(3),3)
2249                ELSE
2250                   USERF = F1 + F5
2251                ENDIF
2252          C
2253                RETURN
```

```
 2254            C
2255            C
2256            C
2257            C     FUNCTION USERF(84)
2258            C     SERVICE TIME OF WIND SHOP FOUR.
2259             8400 NCODE   = ATRIB(18)
2260                  LEVSYS6 = MOD(NCODE,10)
2261            C
2262                  IF(LEVSYS6.GE.4) THEN
2263                     USERF = TRIAG(MIN6(LEVSYS6),MODE6(LEVSYS6),MAX6(LEVSYS6),3)
2264                  ELSE
2265                     USERF = 0.0
2266                  ENDIF
2267            C
2268                  RETURN
2269            C
2270            C
2271            C     END OF SECTION 8, WING SERVICE.
2272            C
2273            C
2274            C
```

```
C     *********
C     SECTION 9, MMT SERVICE  -- USERF(91) TO USERF(96)
C     *********
 9000 CONTINUE
C
C     MMT FIXES ONLY LEVEL 4 AND LEVEL 5 PROBLEMS.
C     GLOSSARY OF VARIABLES:
C        COMMON/UCOM9/MMTMIN(6,2),MMTMOD(6,2),MMTMAX(6,2)
C
C     END OF GLOSSARY
C
      NFUNC = MOD(IFN,10)
      GO TO(9100,9200,9300,9400,9500,9600),NFUNC
C
C
C     FUNCTION USERF(91)
C     SERVICE TIME FOR MMT ONE.
C
 9100 NCODE = ATRIB(18)
      L     = NCODE/100000
      IF(L.GE.4) THEN
         L = L - 3
         USERF = TRIAG(MMTMIN(1,L),MMTMOD(1,L),MMTMAX(1,L),3)
      ELSE
         USERF = 0.0
      ENDIF
C
      RETURN
C
C
C
C     FUNCTION USERF(92)
C     SERVICE TIME FOR MMT TWO.
C
 9200 NCODE = ATRIB(18)
      L     = MOD(NCODE,100000)/10000
      IF(L.GE.4) THEN
         L = L - 3
         USERF = TRIAG(MMTMIN(2,L),MMTMOD(2,L),MMTMAX(2,L),3)
      ELSE
         USERF = 0.0
      ENDIF
C
      RETURN
C
C
C
C     FUNCTION USERF(93)
C     SERVICE TIME FOR MMT THREE.
C
```

```
 2325          9300 NCODE = ATRIB(18)
2326               L     = MOD(NCODE,10000)/1000
2327               IF(L.GE.4) THEN
2328                  L = L - 3
2329                  USERF = TRIAG(MMTMIN(3,L),MMTMOD(3,L),MMTMAX(3,L),3)
2330               ELSE
2331                  USERF = 0.0
2332               ENDIF
2333          C
2334               RETURN
2335          C
2336          C
2337          C    FUNCTION USERF(94)
2338          C    SERVICE TIME FOR MMT FOUR.
2339          C
2340          C
2341           9400 NCODE = ATRIB(18)
2342               L     = MOD(NCODE,1000)/100
2343               IF(L.GE.4) THEN
2344                  L = L - 3
2345                  USERF = TRIAG(MMTMIN(4,L),MMTMOD(4,L),MMTMAX(4,L),3)
2346               ELSE
2347                  USERF = 0.0
2348               ENDIF
2349          C
2350               RETURN
2351          C
2352          C
2353          C
2354          C    FUNCTION USERF(95)
2355          C    SERVICE TIME FOR MMT FIVE.
2356          C
2357           9500 NCODE = ATRIB(18)
2358               L     = MOD(NCODE,100)/10
2359               IF(L.GE.4) THEN
2360                  L = L - 3
2361                  USERF = TRIAG(MMTMIN(5,L),MMTMOD(5,L),MMTMAX(5,L),3)
2362               ELSE
2363                  USERF = 0.0
2364               ENDIF
2365          C
2366               RETURN
2367          C
2368          C
2369          C
2370          C    FUNCTION USERF(96)
2371          C    SERVICE TIME FOR MMT SIX.
2372          C
2373           9600 NCODE = ATRIB(18)
2374               L     = MOD(NCODE,10)
```

```
 2375                IF(L.GE.4) THEN
2376                  L = L - 3
2377                  USERF = TRIAG(MMTMIN(6,L),MMTMOD(6,L),MMTMAX(6,L),3)
2378                ELSE
2379                  USERF = 0.0
2380                ENDIF
2381          C
2382                RETURN
2383          C
2384          C
2385          C     END OF SECTION 7, MMT SERVICE.
2386          C
2387          C
2388          C
```

```
2389            C     *********
2390            C     SECTION 10, SQUADRON SERVICE  --  USERF(101) TO USERF(104)
2391            C     *********
2392           10000 CONTINUE
2393            C
2394            C     GLOSSARY OF VARIABLES:
2395            C        COMMON/UCOMS/MIN1(5),MODE1(5),MAX1(5),MIN2(5),MODE2(5),MAX2(5)
2396            C                    ,MIN3(5),MODE3(5),MAX3(5),MIN4(5),MODE4(5),MAX4(5)
2397            C                    ,MIN5(5),MODE5(5),MAX5(5),MIN6(5),MODE6(5),MAX6(5)
2398            C                    ,WSH1INT(3),WSH3INT(3)
2399            C
2400            C        COMMON/UCOM10/MIN1(5),MODE1(5),MAX1(5)
2401            C
2402            C     FIXES ONLY TWO AND THREE LEVEL PROBLEMS, BUT THE
2403            C     SERVICE IS PERFORMED CONCURRENTLY FOR ALL SHOPS
2404            C     (IE ALL SYSTEMS).  IF TWO SYSTEMS ARE IN THE
2405            C     SAME SHOP, A MAX TIME IS USED, PLUS AN INTERFERENCE TIME
2406            C     FROM ALL OTHER SYSTEMS BEING WORKED ON CONCURRENTLY.
2407            C
2408            C
2409            C
2410                  NFUNC = MOD(IFN,10)
2411                  GO TO(10100,10200,10300,10400),NFUNC
2412            C
2413            C
2414            C     FUNCTION USERF(101)
2415            C     SERVICE TIME FOR SQUADRON SHOP ONE.
2416            C
2417           10100 NCODE = ATRIB(10)
2418                  NINT  = 0
2419                  NDIV  = 1000000
2420            C
2421                  DO 10120 I = 1,6
2422                  NCODE = MOD(NCODE,NDIV)
2423                  NDIV  = NDIV/10
2424                  LEV(I)= NCODE/NDIV
2425                  IF(I.NE.3.AND.I.NE.4.AND.
2426                 &      (LEV(I).EQ.2.OR.LEV(I).EQ.3)) NINT = NINT + 1
2427           10120 CONTINUE
2428            C
2429                  IF(LEV(3).EQ.2.OR.LEV(3).EQ.3) THEN
2430                    F3 = TRIAG(MIN3(LEV(3)),MODE3(LEV(3)),MAX3(LEV(3)),3)
2431                  ELSE
2432                    F3 = 0.0
2433                  ENDIF
2434            C
2435                  IF(LEV(4).EQ.2.OR.LEV(4).EQ.3) THEN
2436                    F4 = TRIAG(MIN4(LEV(4)),MODE4(LEV(4)),MAX4(LEV(4)),3)
2437                  ELSE
2438                    F4 = 0.0
```

```
 2439                ENDIF
2440          C
2441                IF(F3.GT.F4)THEN
2442                   FTEMP = F3
2443                ELSE
2444                   FTEMP = F4
2445                ENDIF
2446          C
2447                IF(NINT.EQ.0.OR.FTEMP.EQ.0.0) THEN
2448                   USERF = FTEMP
2449                ELSE
2450                   USERF = FTEMP + TRIAG(MINI(NINT),MODEI(NINT),MAXI(NINT),3)
2451                ENDIF
2452          C
2453                RETURN
2454          C
2455          C
2456          C     FUNCTION USERF(102)
2457          C     SERVICE TIME FOR SQUADRON SHOP TWO.
2458          C
2459          C
2460          10200 NCODE = ATRIB(18)
2461                NINT  = 0
2462                NDIV  = 1000000
2463          C
2464                DO 10220 I = 1,6
2465                   NCODE = MOD(NCODE,NDIV)
2466                NDIV  = NDIV/10
2467                LEV(I)= NCODE/NDIV
2468          C
2469                   IF(I.NE.2.AND.(LEV(I).EQ.2.OR.LEV(I).EQ.3)) NINT = NINT + 1
2470          10220 CONTINUE
2471          C
2472                IF(LEV(2).EQ.2.OR.LEV(2).EQ.3) THEN
2473                   F2 = TRIAG(MIN2(LEV(2)),MODE2(LEV(2)),MAX2(LEV(2)),3)
2474                ELSE
2475                   F2 = 0.0
2476                ENDIF
2477          C
2478                IF(NINT.EQ.0.OR.F2.EQ.0.0) THEN
2479                   USERF = F2
2480                ELSE
2481                   USERF = F2 + TRIAG(MINI(NINT),MODEI(NINT),MAXI(NINT),3)
2482                ENDIF
2483          C
2484                RETURN
2485          C
2486          C
2487          C     FUNCTION USERF(103)
2488          C     SERVICE TIME FOR SQUADRON SHOP THREE.
```

```
C
10300 NCODE = ATRIB(18)
      NINT  = 0
      NDIV  = 1000000
C
      DO 10320 I = 1,6
         NCODE = MOD(NCODE,NDIV)
         NDIV  = NDIV/10
         LEV(I)= NCODE/NDIV
         IF(I.NE.1.AND.I.NE.5.AND.
     &       (LEV(I).EQ.2.OR.LEV(I).EQ.3)) NINT = NINT + 1
10320 CONTINUE
C
      IF(LEV(1).EQ.2.OR.LEV(1).EQ.3) THEN
         F1 = TRIAG(MIN1(LEV(1)),MODE1(LEV(1)),MAX1(LEV(1)),3)
      ELSE
         F1 = 0.0
      ENDIF
C
      IF(LEV(5).EQ.2.OR.LEV(5).EQ.3) THEN
         F5 = TRIAG(MIN5(LEV(5)),MODE5(LEV(5)),MAX5(LEV(5)),3)
      ELSE
         F5 = 0.0
      ENDIF
C
      IF(F1.GT.F5)THEN
         FTEMP = F1
      ELSE
         FTEMP = F5
      ENDIF
C
      IF(NINT.EQ.0.OR.FTEMP.EQ.0.0) THEN
         USERF = FTEMP
      ELSE
         USERF = FTEMP + TRIAG(MINI(NINT),MODEI(NINT),MAXI(NINT),3)
      ENDIF
C
      RETURN
C
C
C     FUNCTION USERF(104)
C     SERVICE TIME FOR SQUADRON SHOP FOUR.
C
C
C
10400 NCODE = ATRIB(18)
      NINT  = 0
      NDIV  = 1000000
C
      DO 10420 I = 1,6
```

```
 2539                    NCODE = MOD(NCODE,NDIV)
2540                   NDIV  = NDIV/10
2541                   LEV(I)= NCODE/NDIV
2542          C
2543                   IF(I.NE.6.AND.(LEV(I).EQ.2.OR.LEV(I).EQ.3)) NINT = NINT + 1
2544          10420 CONTINUE
2545          C
2546                IF(LEV(6).EQ.2.OR.LEV(6).EQ.3) THEN
2547                   F6 = TRIAG(MIN6(LEV(6)),MODE6(LEV(6)),MAX6(LEV(6)),3)
2548                ELSE
2549                   F6 = 0.0
2550                ENDIF
2551          C
2552                IF(NINT.EQ.0.OR.F6.EQ.0.0) THEN
2553                   USERF = F6
2554                ELSE
2555                   USERF = F6 + TRIAG(MINI(NINT),MODEI(NINT),MAXI(NINT),3)
2556                ENDIF
2557          C
2558                RETURN
2559          C
2560          C
2561          C
2562          C
2563          C     END OF SECTION 10, SQUADRON SERVICE.
2564          C
2565          C
2566          C
```

```
2567            C     *********
2568            C     SECTION 11, MAINTENANCE CONTROL -- USERF(111) TO USERF(113)
2569            C     *********
2570            11000 CONTINUE
2571            C
2572            C     GLOSSARY OF VARIABLES USED IN THIS SECTION
2573            C
2574            C     SLAM FUNCTIONS:
2575            C        NNRSC  - DETERMINE AMOUNT OF RESOURCE AVAILABLE
2576            C          (1)    - WGSHOP1 RESOURCE
2577            C          (2)    - WGSHOP2 RESOURCE
2578            C          (3)    - WGSHOP3 RESOURCE
2579            C          (4)    - WGSHOP4 RESOURCE
2580            C          (5)    - MMT1    RESOURCE
2581            C          (6)    - MMT2    RESOURCE
2582            C          (7)    - MMT3    RESOURCE
2583            C          (8)    - MMT4    RESOURCE
2584            C          (9)    - MMT5    RESOURCE
2585            C          (10)   - MMT6    RESOURCE
2586            C
2587            C     INTEGER VALUE FUNCTIONS:
2588            C        NFUNC  - DETERMINE WHICH FUNCTION TO ACCESS.
2589            C        N      - CODED MX FAILURE CODE.
2590            C
2591            C     END OF GLOSSARY.
2592            C
2593                  NFUNC = MOD(IFN,10)
2594                  GO TO (11100,11200,11300),NFUNC
2595            C
2596            C
2597            C     FUNCTION USERF(111)
2598            C     DETERMINE IF ENTITY REQUIRES WING SERVICE AND IF IT'S AVAILABLE.
2599            C     IF SO, RETURN 1, ELSE RETURN 0.
2600            C
2601            11100 USERF = 0.0
2602                  N     = ATRIB(18)
2603                  IF(N/100000.GE.4.AND.NNRSC(3).GT.0.OR.
2604                 +  MOD(N,100000)/10000.GE.4.AND.NNRSC(2).GT.0.OR.
2605                 +  MOD(N,10000)/1000  .GE.4.AND.NNRSC(1).GT.0.OR.
2606                 +  MOD(N,1000)/100    .GE.4.AND.NNRSC(1).GT.0.OR.
2607                 +  MOD(N,100)/10      .GE.4.AND.NNRSC(3).GT.0.OR.
2608                 +  MOD(N,10)          .GE.4.AND.NNRSC(4).GT.0) USERF = 1.0
2609                  RETURN
2610            C
2611            C
2612            C     FUNCTION USERF(112)
2613            C     DETERMINE IF A/C REQUIRES MMT IF IT IS AVAILABLE.
2614            C     IF SO, RETURN 1, ELSE RETURN 0.
2615            C
2616            11200 USERF = 0.0
```

```
 2617                N     = ATRIB(18)
2618                 IF(N/100000          .GE.4.AND.NNRSC(5).GT.0.OR.
2619                +  MOD(N,100000)/10000.GE.4.AND.NNRSC(6).GT.0.OR.
2620                +  MOD(N,10000)/1000  .GE.4.AND.NNRSC(7).GT.0.OR.
2621                +  MOD(N,1000)/100    .GE.4.AND.NNRSC(8).GT.0.OR.
2622                +  MOD(N,100)/10      .GE.4.AND.NNRSC(9).GT.0.OR.
2623                +  MOD(N,10)          .GE.4.AND.NNRSC(10).GT.0) USERF = 1.0
2624                 RETURN
2625           C
2626           C
2627           C     FUNCTION USERF(113)
2628           C     DETERMINE IF OPENING AVAILABLE IN WING MX FOR A/C WAITING ON MMT.
2629           C
2630           11300 N     = ATRIB(18)
2631                 USERF = 0.0
2632                 IF(XX(93).EQ.0.AND.
2633                +  (NNRSC(3).GT.0.AND. N/100000.GE.4 .OR.
2634                +   NNRSC(2).GT.0.AND. MOD(N,100000)/10000.GE.4.OR.
2635                +   NNRSC(1).GT.0.AND. MOD(N,10000)/1000.GE.4.OR.
2636                +   NNRSC(1).GT.0.AND. MOD(N,1000)/100.GE.4.OR.
2637                +   NNRSC(3).GT.0.AND. MOD(N,100)/10.GE.4.OR.
2638                +   NNRSC(4).GT.0.AND. MOD(N,10).GE.4)) USERF = 1.0
2639                 RETURN
2640           C
2641           C
2642           C     END OF SECTION 11, MAINTENANCE CONTROL.
2643           C
```

```
C     *********
C     SECTION 12, STATISTICS -- USERF(121) TO USERF(125)
C     *********
C
12000 CONTINUE
C
      NFUNC = MOD(IFN,10)
      GO TO(12100,12200,12300,12400,12500),NFUNC
C
C     FUNCTION USERF(121)
C     INCREMENT NUMBER OF SORTIES FOR A/C AND PILOT BY DAY.
C
C
12100 IF(TNOW.LT.1440.0) THEN
         ATRIB(4) = ATRIB(4) + 1
         ATRIB(34)= ATRIB(34)+ 1
      ELSEIF(TNOW.LT.2880.0) THEN
         ATRIB(5) = ATRIB(5) + 1
         ATRIB(35)= ATRIB(35)+ 1
      ELSE
         ATRIB(6) = ATRIB(6) + 1
         ATRIB(36)= ATRIB(36)+ 1
      ENDIF
C
      USERF = 0.0
      RETURN
C
C
C     FUNCTION USERF(122)
C     MARK CORRECT ATRIB (1 PER DAY) WITH TNOW SO PILOT FLYING TIME
C     STATISTICS CAN BE COLLECTED.
C
12200 IF(TNOW.LT.1440.0) THEN
         ATRIB(37) = TNOW
      ELSEIF(TNOW.LT.2880.0) THEN
         ATRIB(38) = TNOW
      ELSE
         ATRIB(39) = TNOW
      ENDIF
C
      USERF = 0.0
      RETURN
C
C
C     FUNCTION USERF(123)
C     MARK CORRECT ATRIB (BY DAY) SO TIME BETWEEN PILOT SEPERATION AND
C     NEXT FLIGHT ON SAME DAY CAN BE CALCULATED.
C
12300 IF(TNOW.LT.1440.0) THEN
         ATRIB(10) = TNOW
```

```
 2694                ELSEIF(TNOW.LT.2880.0) THEN
2695                  ATRIB(11) = TNOW
2696                ELSE
2697                  ATRIB(12) = TNOW
2698                ENDIF
2699          C
2700                USERF = 0.0
2701                RETURN
2702          C
2703          C
2704          C     FUNCTION USERF(124)
2705          C     CALCULATE TOTAL A/C OPERATING TIME PER DAY
2706          C
2707          12400 IF(TNOW.LT.1440.0) THEN
2708                  ATRIB(28) = ATRIB(7)
2709                ELSEIF(TNOW.LT.2880.0) THEN
2710                  ATRIB(29) = ATRIB(7) - ATRIB(28)
2711                ELSE
2712                  ATRIB(30) = ATRIB(7) - ATRIB(28) - ATRIB(29)
2713                ENDIF
2714          C
2715                USERF = 0.0
2716                RETURN
2717          C
2718          C
2719          C     FUNCTION USERF(125)
2720          C     MARK CORRECT ATRIB (1 PER DAY) TO TNOW FOR MX AND SERVICE
2721          C     TURNAROUND TIME.
2722          C
2723          12500 IF(TNOW.LT.1440.0) THEN
2724                  ATRIB(25) = TNOW
2725                ELSEIF(TNOW.LT.2880.0) THEN
2726                  ATRIB(26) = TNOW
2727                ELSE
2728                  ATRIB(27) = TNOW
2729                ENDIF
2730          C
2731                USERF = 0.0
2732                RETURN
2733          C
2734          C
2735          C     END OF SECTION 12, STATISTICS.
2736          C
2737          C
2738          C
2739          C
```

```
2740            C     *********
2741            C     SECTION 13, MX FAILURE DISTRIBTIONS -- USERF(131) TO USERF(137)
2742            C     *********
2743            C
2744            13000 CONTINUE
2745            C
2746                  NFUNC = MOD(IFN,10)
2747                  GOTO(13100,13100,13100,13100,1310w,13100,13700),NFUNC
2748            C
2749            C     FUNCTION USERF(131) TO FUNCTION USERF(136)
2750            13100 USERF=(1+BET(NFUNC)/ALP(NFUNC))*MTBF(NFUNC)*
2751                 &      BETA(ALP(NFUNC),BET(NFUNC),2)
2752                  RETURN
2753            C
2754            C
2755            C
2756            C     FUNCTION USERF(137)
2757            C     UNIFORMLY DISTRIBUTE THE INITIAL ENGINE RUN TIME.
2758            13700 USERF = UNFRM(ERUNMIN,ERUNMAX,2)
2759                  RETURN
2760            C
2761            C
2762            C     END OF SECTION 13.
2763            C
2764                  END
2765            C
2766            C
2767            C
2768            C
2769            C
2770            C
2771            C
2772            C
```

```
 2773                SUBROUTINE EVENT(IEV)
2774                COMMON/SCOM1/ATRIB(100),DD(100),DDL(100),DTNOW,II,MFA,MSTOP,NCLNR
2775               &,NCRDR,NPRNT,NNRUN,NNSET,NTAPE,SS(100),SSL(100),TNEXT,TNOW,XX(100)
2776                COMMON QSET(1)
2777                COMMON/PFLAG/MAXPRT,LEVPRT,BPRT,EPRT
2778          C
2779                COMMON/SCHEDLR/NCAGLE(3,2),DELAY(3,2),SRATE(3)
2780                COMMON/MSNREQ /INITAC(3),REQPCT(3),LIMITAC,ACTIVE(6)
2781                COMMON/STATS  /MAXCONF(6),NACTYPE(6,3),NCSQ(300),MSNRQA2,MSNFLW(3)
2782                COMMON/LOCAL  /PRIORTY(6),NCAGCNT(2)
2783                COMMON/CLOCK  /INTDARK,DUSK(3),DAYLGHT(3),MAJEVNT(13,2)
2784          C
2785                COMMON/UCOM2/NPARK(6,50,2),NTYPE(6,3)
2786          C
2787                REAL A(50),MAJEVNT
2788                INTEGER PRIORTY,NAVAIL(6),NSET(1),LOSTFTL(6)
2789                EQUIVALENCE (NSET(1),QSET(1))
2790                LOGICAL FORMED,REQSUPY(6),DAYTIME,ACTIVE,INTDARK
2791          C
2792          C     NIGHT/DAY CLOCK
2793                IF(INTDARK)THEN
2794                   DAYTIME = .FALSE.
2795                   DO 10 I = 1,3
2796                      IF(TNOW.GE.DAYLGHT(I).AND.TNOW.LT.DUSK(I))DAYTIME=.TRUE.
2797            10    CONTINUE
2798                ELSE
2799                   DAYTIME = .TRUE.
2800                   DO 20 I = 1,3
2801                      IF(TNOW.GE.DUSK(I).AND.TNOW.LT.DAYLGHT(I))DAYTIME=.FALSE.
2802            20    CONTINUE
2803                ENDIF
2804          C
2805          C     DAY OF CONFLICT CLOCK
2806                IF(TNOW.LT.1440.0) THEN
2807                   NDAY = 1
2808                ELSEIF(TNOW.LT.2880.0) THEN
2809                   NDAY = 2
2810                ELSE
2811                   NDAY = 3
2812                ENDIF
2813          C
2814          C
2815          C
2816          C
2817          C
2818                IF(LEVPRT.GE.5.AND.TNOW.GE.BPRT.AND.TNOW.LE.EPRT)THEN
2819                   PRINT*,'   EVENT ',IEV,' CALLED, TIME ',TNOW,', A/C ',ATRIB(2)
2820                ENDIF
2821          C
2822                IF(IEV.EQ.10)GOTO 1000
```

```
 2823                IF(IEV.EQ.11)GOTO 1100
2824                GO TO (100,200,300,400,500,600,700),IEV
2825          C
2826          C
```

```
C     *********
C     MAJOR EVENT CLOCK  --  EVENT 1.
 100  CONTINUE
C     *********
C
C
C     MAJOR EVENT CODES:
C        0 - TERMINATE
C        1 - SCHEDULER
C        2 - NIGHT PARK
C        3 - GRA SWITCH
C        4 - RESUPPLY/RECONFIURE
C        99- NOTHING
C
C     MAJOR EVENTS MUST BE STORED "IN ORDER" IN MAJEVNT(J,1), WITH TYPE
C        OF EVENT IN MAJEVNT(J,2).  MAXIUM OF 13 EVENTS.
C
      XX(98) = 999999.0
      XX(96) = 0.0
      IF(TNOW.LT.MAJEVNT(1,1))THEN
C        IF EVENT 1 IS CALLED PRIOR TO THE FIRST ACTIVITY, DELAY
C        UNTIL THE FIRST ACTIVITY SHOULD BE CALLED.
         XX(98) = MAJEVNT(1,1) - TNOW
         XX(96) = 0.0
      ELSE
C     FIND THE ACTIVITY BEING CALLED, AND RELEASE THE ENTITY TO
C     INITIATE THAT ACTIVITY.
      DO 110 I = 1,13
      IF(TNOW.EQ.MAJEVNT(I,1).AND.MAJEVNT(I,2).NE.99.0)THEN
         IF(MAJEVNT(I,2).GT.0.0.AND.MAJEVNT(I,2).LE.4.0)THEN
            XX(96) = MAJEVNT(I,2)
            IF(I.NE.13)XX(98) = MAJEVNT(I+1,1) - TNOW
         ELSE
            XX(96) = 0.0
         ENDIF
         MAJEVNT(I,2) = 99.0
      ENDIF
 110  CONTINUE
      ENDIF
      RETURN
C
```

```
C     *********
C     INITIALIZE SCHEDULER  --  EVENT 2.
C     *********
 200  CONTINUE
C
C
C     SET INITIAL ORDER OF SQDNS TO FORM A FLIGHT FROM.
      DO 210 I = 1,6
         PRIORTY(I) = I
 210  CONTINUE
C
      NTOTAC = 0
      DO 220 I = 1,300
         IF(NCSQ(I).GT.0.AND.NCSQ(I).LE.6)NTOTAC = NTOTAC + 1
 220  CONTINUE
      DO 230 I = 1,6
         IF(ACTIVE(I))NTOTAC = NTOTAC - XX(61)
 230  CONTINUE
      MSNRQA2 = (NTOTAC*SRATE(NDAY)*REQPCT(2)+1.5)/3.0
C
      IF(LEVPRT.GE.3.AND.TNOW.GE.BPRT.AND.TNOW.LE.EPRT)THEN
         PRINT*,'-NTOTAC IS ',NTOTAC,' AND MSNRQA2 IS ',MSNRQA2
      ENDIF
C
C     RESET SCHEDULER COUNTERS.
      NGAGCNT(1) = 0
      NGAGCNT(2) = 0
      MSNFLW(1) = 0
      MSNFLW(2) = 0
      MSNFLW(3) = 0
C
      IF(LEVPRT.GE.2.AND.TNOW.GE.BPRT.AND.TNOW.LE.EPRT)THEN
         PRINT*,'***  INITIALIZE SCHEDULER, TIME ',TNOW
      ENDIF
C
      RETURN
C
C
C
C     *********
C     SCHEDULER  --  EVENT 3.
C     *********
 300  CONTINUE
C
C
      XX(73) = 0.0
      XX(96) = 0.0
      FORMED = .FALSE.
C
C
```

```
2916                IF(DAYTIME) THEN
2917          C
2918          C      FIRST FORM ALL THE FLIGHTS FOR CAGGLE NUMBER 1 OF NDAY
2919          C      (OR AS MANY FLIGHTS AS THERE ARE AIRCRAFT).  THEN SCHEDULE
2920          C      THE DELAY FOLLOWING THE LAUNCH OF THE FIRST CAGGLE.
2921          C
2922                 IF(NCAGCNT(1).LT.NCAGLE(NDAY,1)) THEN
2923                    NCAGCNT(1) = NCAGCNT(1) + 1
2924                    XX(48) = 3.0
2925                    XX(97) = 3.0
2926                    CALL ORGANFT(PRIORTY,XX(48),NDAY,FORMED,NSQ,XX(49),MFA)
2927                    XX(96) = NSQ
2928                    IF(.NOT.FORMED) THEN
2929                       NCAGCNT(1) =999999
2930                       XX(98) = DELAY(NDAY,1)
2931                       XX(96) =8.0
2932                    ENDIF
2933          C
2934                    IF(LEVPRT.GE.4.AND.TNOW.GE.BPRT.AND.TNOW.LE.EPRT)THEN
2935                    IF(FORMED) PRINT*,'CAGGLE 1:FLT ',NCAGCNT(1),'OF DAY ',NDAY
2936                    ENDIF
2937          C
2938                 ELSEIF(NCAGCNT(1).EQ.NCAGLE(NDAY,1).AND.
2939               &                    NCAGLE(NDAY,1).NE.0)THEN
2940                    NCAGCNT(1) = 999999
2941                    XX(98) = DELAY(NDAY,1)
2942                    XX(96) = 8.0
2943          C
2944                    IF(LEVPRT.GE.4.AND.TNOW.GE.BPRT.AND.TNOW.LE.EPRT)THEN
2945                       PRINT*,'DELAY AFTER CAGGLE 1 ON DAY ',NDAY
2946                    ENDIF
2947          C
2948                 ELSEIF(NCAGCNT(2).LT.NCAGLE(NDAY,2)) THEN
2949          C
2950          C         NEXT FORM ALL THE FLIGHTS REQUIRED FOR THE SECOND CAGGLE
2951          C         FOLLOWED BY ANOTHER DELAY.  IF ENOUGH AIRCRAFT ARE NOT
2952          C         AVAILABLE, THEN ONLY THE NUMBER OF POSSIBLE FLIGHTS ARE
2953          C         SENT.
2954          C
2955                    NCAGCNT(2) = NCAGCNT(2) + 1
2956                    XX(48) = 3.0
2957                    XX(97) = 3.0
2958                    CALL ORGANFT(PRIORTY,XX(48),NDAY,FORMED,NSQ,XX(49),MFA)
2959                    XX(96) = NSQ
2960                    IF(.NOT.FORMED) THEN
2961                       NCAGCNT(2) = 999999
2962                       XX(98) = DELAY(NDAY,2)
2963                       XX(96) = 8.0
2964                    ENDIF
2965          C
2966
2967
```

```
2968                      IF(LEVPRT.GE.4.AND.TNOW.GE.BPRT.AND.TNOW.LE.EPRT)THEN
2969                    IF(FORMED) PRINT*,'CAGGLE 2:FLT ',NCAGCNT(2),'OF DAY ',NDAY
2970                    ENDIF
2971          C
2972                ELSEIF(NCAGCNT(2).EQ.NCAGLE(NDAY,2).AND.
2973               &                    NCAGLE(NDAY,2).NE.0)THEN
2974                    NCAGCNT(2) = 999999
2975                    XX(98) = DELAY(NDAY,2)
2976                    XX(96) = 8.0
2977          C
2978                    IF(LEVPRT.GE.4.AND.TNOW.GE.BPRT.AND.TNOW.LE.EPRT)THEN
2979                       PRINT*,'DELAY AFTER CAGGLE 2 ON DAY ',NDAY
2980                    ENDIF
2981          C
2982                ELSE
2983          C
2984          C        FOLLOWING THE SCHEDULING OF THE CAGGLES TO AREA THREE,
2985          C        SCHEDULE FLIGHTS FOR AREA 2 AS LONG AS POSSIBLE OR AS
2986          C        MANY AS REQUIRED (MSNRQA2).  ONCE NO MORE AREA 2 MISSIONS
2987          C        CAN BE SCHEDULED, AREA 1 MISSIONS ARE SCHEDULED UNTIL
2988          C        NO MORE FLIGHTS CAN BE FORMED.  WHEN AN ARCRAFT RETURNS
2989          C        TO THE READY POOL, ANOTHER FLIGHT IS FORMED, IF POSSIBLE.
2990          C        IF IT CAN BE FORMED, IT IS SENT TO THE CORRECT AREA,
2991          C        AREA 2 IF MSNRQA2 IS NOT MET, ELSE AREA 1 (THIS ALSO
2992          C        DEPENDS ON THE TANK CONFIGURATION OF THE AIRCRAFT).
2993          C
2994                    IF(MSNFLW(2).LT.MSNRQA2) THEN
2995                       XX(97) = 2.0
2996                       XX(48) = 2.0
2997                       CALL ORGANPT(PRIORTY,XX(48),NDAY,FORMED,NSQ,XX(49),MFA)
2998                       IF(.NOT.FORMED) THEN
2999                       XX(48) = 3.0
3000                       CALL ORGANPT(PRIORTY,XX(48),     FORMED,NSQ,XX(49),MFA)
3001                       ENDIF
3002                       XX(96) = NSQ
3003                    ENDIF
3004          C
3005                    IF(.NOT.FORMED) THEN
3006                       XX(97) = 1.0
3007                       XX(48) = 1.0
3008                       CALL ORGANPT(PRIORTY,XX(48),NDAY,FORMED,NSQ,XX(49),MFA)
3009                       IF(.NOT.FORMED) THEN
3010                       XX(48) = 2.0
3011                       CALL ORGANPT(PRIORTY,XX(48),NDAY,FORMED,NSQ,XX(49),MFA)
3012                       ENDIF
3013                       IF(.NOT.FORMED) THEN
3014                       XX(48) = 3.0
3015                       CALL ORGANPT(PRIORTY,XX(48),NDAY,FORMED,NSQ,XX(49),MFA)
3016                       ENDIF
3017                       XX(96) = NSQ
```

```
 3018                     ENDIF
3019          C
3020                     IF(.NOT.FORMED) XX(96) = 7.0
3021                  ENDIF
3022          C
3023                  IF(FORMED) THEN
3024                     XX(47) = XX(47) + 1
3025                     IF(XX(47).GT.46) XX(47) = 1
3026                     NACTYPE(NSQ,IFIX(XX(48)))=NACTYPE(NSQ,IFIX(XX(46))) - 3
3027                  ENDIF
3028          C
3029                  IF(LEVPRT.GE.3.AND.TNOW.GE.BPRT.AND.TNOW.LE.EPRT)THEN
3030                  IF(FORMED) THEN
3031                     PRINT*,'-MSN',XX(47),' GOING TO',XX(97),' WITH TANKS',XX(48)
3032                  ELSE
3033                     PRINT*,' NOT FORMED'
3034                  ENDIF
3035                  ENDIF
3036          C
3037               ELSE
3038          C
3039          C       THE ABOVE SCHEDULING PROCEDURES ARE FOLLOWED UNTIL NIGHT
3040          C       FALL, WHEN THE SCHEDULING OF MISSIONS IS ENDED.
3041          C
3042                  XX(96) = 0.0
3043          C
3044               ENDIF
3045          C
3046          C
3047          C     ONCE THE QUOTAS FOR AREA 3 AND AREA 2 MISSIONS ARE MET,
3048          C     CHANGE THE SPECIFIED CONFIGURATION OF A SQUADRON SO
3049          C     SUBROUTINE RECONFIGURE, USERF(73), CAN MAKE INTELLIGENT
3050          C     DECISIONS ON WHETHER OR NOT TO RECONFIGURE AN AIRCRAFT.
3051          C
3052               IF(MSNFLW(2).GE.MSNRQA2)THEN
3053                  DO 320 I = 1,6
3054                     MAXCONF(I) = 1
3055           320    CONTINUE
3056               ELSEIF(NGAGLE(NDAY,1).EQ.0.AND.NGAGLE(NDAY,2).EQ.0.OR.
3057              &   NGAGCNT(1).EQ.999999.AND.NGAGCNT(2).EQ.0.OR.
3058              &                                NGAGCNT(2).EQ.999999)THEN
3059                  DO 340 I = 1,6
3060                     IF(MAXCONF(I).EQ.3)MAXCONF(I) = 2
3061           340    CONTINUE
3062               ENDIF
3063          C
3064               RETURN
3065          C
3066          C
3067          C
```

```
3068            C     *********
3069            C     INITIAL NIGHT PARK  --  EVENT 4.
3070            C     *********
3071             400  CONTINUE
3072            C
3073            C
3074            C     PERFORM THE INITIAL NIGHT PARKING.  MOVE ALL AIRCRAFT NOT
3075            C     PARKED IN A SHELTER TO A SHELTER (TWO A/C TO A SHELTER,
3076            C     EXCEPT QRA SHELTERS).  THIS RULE APPLIES TO ALL AIRCRAFT
3077            C     EXCEPT THOSE IN MAINTENANCE.  WHEN A MAINTENANCE
3078            C     AIRCRAFT RETURNS TO THE SQUADRON AREA, THE NEXT ROUTINE -
3079            C     EVENT 5 - TRIES TO PARK IT IN A SHELTER.
3080            C
3081                  CALL NIGHTPK(1,NPARK,NTYPE)
3082            C
3083                  IF(LEVPRT.GE.2.AND.TNOW.GE.BPRT.AND.TNOW.LE.EPRT) THEN
3084                    PRINT*,'***  INITIAL NIGHT PARK, TIME ',TNOW
3085                  ENDIF
3086            C
3087                  RETURN
3088            C
3089            C
3090            C
3091            C     *********
3092            C     FOLLOW-ON NIGHT PARKING  --  EVENT 5.
3093            C     *********
3094             500  CONTINUE
3095            C
3096            C
3097                  XX(96) = 0.0
3098                  IF(.NOT.DAYTIME) THEN
3099                    XX(96) = 1.0
3100                    CALL NIGHTPK(2,NPARK,NTYPE)
3101                  ENDIF
3102            C
3103                  IF(LEVPRT.GE.3.AND.TNOW.GE.BPRT.AND.TNOW.LE.EPRT) THEN
3104                  PRINT*,'-FINISHED NIGHTPK(2)'
3105                  ENDIF
3106            C
3107                  RETURN
3108            C
3109            C
3110            C
```

```
C     *********
C     QRA CHANGE OVER  --  EVENT 6.
C     *********
 600  CONTINUE
C
C
C     CHANGE OVER OF THE QRA PILOTS FOR ALL THE SQUADRONS.  ALSO,
C     IF A REPLACEMENT SQUADRON ARRIVED DURING THE DAY, TAKE THE
C     OLD SQUADRONS AIRCRAFT OFF QRA AND DISPERSE, AND PUT THE
C     REPLACEMENTS AIRCRAFT ON QRA.
C
      CALL QRASWCH(NCSQ,NPARK)
C
      IF(LEVPRT.GE.2.AND.TNOW.GE.BPRT.AND.TNOW.LE.EPRT) THEN
         PRINT*,'***  QRA CHANGE OVER,   TIME ',TNOW
      ENDIF
C
      RETURN
C
C
C
```

```
C     *********
C     RESUPPLY AND RECONFIGURATION  --  EVENT 7.
C     *********
 700  CONTINUE
C
C
 700  DO 710 I = 1,6
        LOSTFTL(I) = 0
        NAVAIL(I) = 0
 710  CONTINUE
C
C     DISPLAY THE JUNK FILE (IF SPECIFIED) AND CALCULATE THE NUMBER
C     OF FLIGHT LEADS LOST PER SQUADRON.
C
      IF(LEVPRT.GE.1.AND.TNOW.GE.BPRT.AND.TNOW.LE.EPRT) THEN
        PRINT*,'******      ***'
        PRINT*,'  JUNK FILE'
      ENDIF
C
      IF(NNQ(99).NE.0) THEN
        IF(LEVPRT.GE.1.AND.TNOW.GE.BPRT.AND.TNOW.LE.EPRT)THEN
          PRINT*,'  SQDN    TAIL    FAILURE    BATTLE    PILOT '
        ENDIF
        DO 720 I = 1,NNQ(99)
          L = LOCAT(I,99)
C
          IF(LEVPRT.GE.1.AND.TNOW.GE.BPRT.AND.TNOW.LE.EPRT) THEN
            PRINT 711,QSET(L+1),QSET(L+2),QSET(L+18),QSET(L+16),
     &                QSET(L+32)
          ENDIF
 711      FORMAT(' ',5(F7.0,1X))
C
          NSQ = QSET(L+1)
          NPL = QSET(L+32)
          NST = QSET(L+33)
          IF(NPL.NE.0.AND.NST.GT.0)LOSTFTL(NSQ) = LOSTFTL(NSQ) + 1
 720    CONTINUE
      ENDIF
C
      IF(LEVPRT.GE.3.AND.TNOW.GE.BPRT.AND.TNOW.LE.EPRT) THEN
      PRINT*,'-FLT LEADS LOST BY SQDN: ',(LOSTFTL(I),I=1,6)
      ENDIF
C
      CALL RESUPLY(ACTIVE,NCSQ,LOSTFTL,LIMITAC,REQSUPY,TNOW)
C
      IF(LEVPRT.GE.2.AND.TNOW.GE.BPRT.AND.TNOW.LE.EPRT) THEN
        PRINT*,'***  RESUPPLY DETERMINED, TIME ',TNOW
      ENDIF
C
C     SET NETWORK VARIABLES TO 1 IF SQUADRON REQUIRES RESUPPLY,
```

```
C     SET TO 0 IF NO RESUPPLY IS REQUIRED.
C
      DO 730 I = 1,6
         XX(49+I) = 0.0
         IF(ACTIVE(I).AND.REQSUPY(I)) XX(49+I) = 1.0
 730  CONTINUE
C
C
C
C     DETERMINE THE NUMBER OF AIRCRAFT AVAILABLE TO FLY A MISSION
C     PER SQUADRON, THEN RECONFIGURE THE AIRCRAFT BASED ON THE
C     SAME RATIOS AS ORIGINALLY DESIGNATED (INITAC).
C
      DO 740 I = 1,300
      IF(NCSQ(I).GT.0.AND.NCSQ(I).LE.6) THEN
         NSQ = NCSQ(I)
         NAVAIL(NSQ) = NAVAIL(NSQ) + 1
      ENDIF
 740  CONTINUE
C
      IF(NNQ(19).GT.0) THEN
      DO 750 I = 1,NNQ(19)
         NTAIL = QSET(LOCAT(I,19)+2)
         NSQ   = NCSQ(NTAIL)
         NAVAIL(NSQ) = NAVAIL(NSQ)-1
 750  CONTINUE
      ENDIF
C
      IF(LEVPRT.GE.2.AND.TNOW.GE.BPRT.AND.TNOW.LE.EPRT) THEN
      PRINT*,'-A/C AVAILABLE BY SQDN: ',(NAVAIL(I),I=1,6)
      ENDIF
C
      CALL RECONFG(INITAC,MAXCONF,NACTYPE,NAVAIL,TNOW)
C
      IF(LEVPRT.GE.2.AND.TNOW.GE.BPRT.AND.TNOW.LE.EPRT) THEN
         PRINT*,'***  A/C RECONFIGURED, TIME ',TNOW
      ENDIF
C
C
      RETURN
C
C
C
```

```
C     **********
C     SPARE AN AIRCRAFT  --  EVENT 10.
C     **********
 1000 CONTINUE
C
C
C     SEE IF A SPARE A/C IS AVAILABLE, IF SO SCHEDULE IT TO ARRIVE
C     AT PREFLIGHT AFTER A SMALL DELAY (TIME TO SWITCH PILOTS, ETC.)
      XX(95)=0.0
      NSQ=ATRIB(1)
      NTK=ATRIB(12)
      NRP=NSQ*3-2
C
      IF(NNQ(NRP).GT.0)THEN
        J=0
 1010   IF(J.EQ.NNQ(NRP))GOTO 1030
        J = J + 1
        IF(QSET(LOCAT(J,NRP)+12).NE.NTK)GOTO 1010
C
        L = LOCAT(J,NRP)
        IF(LEVPRT.GE.4.AND.TNOW.GE.BPRT.AND.TNOW.LE.EPRT) THEN
          PRINT*,' SPARE A/C FOUND'
          PRINT*,' TAIL ',QSET(L+2),', ASSIGNED TO MSN ',ATRIB(46)
        ENDIF
C
        XX(95) = 1.0
        CALL RMOVE(J,NRP,A)
        DO 1020 I = 31,47
          A(I) = ATRIB(I)
 1020   CONTINUE
      A(48) = 0.0
        NACTYPE(NSQ,NTK) = NACTYPE(NSQ,NTK) - 1
        CALL SCHDL(11,TRIAG(4.,5.,6.,3),A)
 1030   CONTINUE
      ENDIF
C
      RETURN
C
C
C
C     **********
C     FILE SPARE A/C AT MX PREFLIGHT  --  EVENT 11.
C     **********
C
C
 1100 CALL FILEM(21,ATRIB)
      RETURN
C
      END
C
```

```
3275              C
3276              C
3277              C
3278              C
```

```
      SUBROUTINE ORGANPT(PRIORTY,TANKS,DAY,FORMED,NSQ,FLTLDS,MFA)
      COMMON QSET(1)
      COMMON/PFLAG/MAXPRT,LEVPRT,BPRT,EPRT
C
      INTEGER POINT(100),CREW(3),FIRST,DAY,PILOT,PRIORTY(6),ACPOOL,
     &       NSET(1),PLPOOL
      LOGICAL FORMED
      EQUIVALENCE (NSET(1),QSET(1))
C
C     PRIORTY IS A QUEUE OF SQUADRON NUMBERS.  THE FIRST SQUADRON TO BE
C     CHECKED TO FORM A FLIGHT IS PRIORTY(1), AND SO ON TO PRIORTY(6).
C
C     FORMED IS A LOGICAL VARIABLE (TRUE OR FALSE) WHICH TELLS THE
C     EXECUTIVE ROUTINE IF THE FUNCTION WAS SUCCESSFUL IN FORMING A FLT.
C
C     TANKS IS THE REQ'D TANK CONFIGURATION FOR THE FLIGHT.  ALL A/C
C     SHOULD HAVE THE SAME TANK CONFIGURATION.
C
C     NSQ IS THE CURRENT SQUADRON BEING EXAMINED, AND THE ONE SELECTED
C      IF THE ROUTINE CAN ORGANIZE THE PILOTS.
C
C
C
C     FLIGHTS ARE FORMED BY THE FOLLOWING RULES:
C     1.  A SQUADRON IS FOUND THAT HAS AT LEAST THREE PILOTS AND
C         AIRCRAFT OF THE DESIRED TANK CONFIGURATION.
C     2.  WITHIN THAT SQUADRON, A FLIGHT LEAD QUALIFIED PILOT IS
C         FOUND.  IF NO FLIGHT LEAD QUALIFIED PILOT IS FOUND,
C         ANOTHER SQUADRON IS SELECTED.
C     3.  NOW TRY TO FIND A SECOND FLIGHT LEAD QUALIFIED PILOT.  IF
C         ONE IS FOUND, THE PILOT GETS A/C 3 IN THE FLIGHT.  IF
C         ONE IS NOT LOCATED, GET A NON-FLIGHT LEAD QUALIFIED PILOT
C         AND PLACE HIM IN A/C 3.
C     4.  FOR THE SECOND AIRCRAFT, GET A NON-QUALIFIED PILOT.
C         IF NONE ARE AVAILABLE, USE A FLIGHT LEAD QUALIFIED PILOT.
C     5.  IF THE FLIGHT CAN NOT BE FORMED FROM THIS SQUADRON, TRY
C         THE NEXT SQUADRON UNTIL ALL HAVE BEEN EXAMINED OR A
C         FLIGHT CAN BE FORMED.
C
C     THE ABOVE RULES IMPLY THAT A CASE 2 FLIGHT IS PREFERRED,
C     WITH A CASE 1 OR CASE 3 BEING FORMED IF CASE 2 IS IMPOSSIBLE.
C
C     THE CASE DEFINITIONS ARE:
C          FLD - FLIGHT LEAD QUALIFIED
C          NFL - NON FLIGHT LEAD QUALIFIED
C
C          CASE     A/C 1     A/C 2     A/C 3
C            1        FLD       NFL       NFL
C            2        FLD       NFL       FLD
C            3        FLD       FLD       FLD
```

```
3329            C
3330            C
3331            C     ADDITIONAL RESTRICTIONS ARE ALSO PLACED ON THE SELECTION
3332            C     OF A PARTICULAR PILOT IF ONE OR MORE PILOTS IN A SQUADRON
3333            C     READY POOL MEET THE ABOVE CRITERIA.  THE TIE BREAKING RULES
3334            C     ARE:
3335            C     1.  SELECT THE FIRST QUALIFIED PILOT IN THE READY POOL THAT
3336            C         HAS FLOWN LESS THAN THREE SORTIES FOR THE DAY.
3337            C     2.  IF ALL HAVE FLOWN THREE OR MORE SORTIES FOR THE DAY,
3338            C         SELECT THE FIRST ONE IN THE READY POOL.
3339            C
3340            C
3341            C
3342            C
3343            C
3344            C     SET UP INITIAL PARAMETERS.
3345            C
3346                  FORMED = .FALSE.
3347            C
3348                  J = 1
3349             100  CONTINUE
3350            C
3351            C        SELECT THE SQUADRON TO BE EXAMINED.
3352                     NSQ    = PRIORTY(J)
3353                     ACPOOL = NSQ*3 - 2
3354                     PLPOOL = NSQ*3 - 1
3355                     NWTANKS= 0
3356            C
3357            C        COUNT THE NUMBER OF AIRCRAFT IN THE SELECTED SQUADRON WITH
3358            C        THE CORRECT CONFIGURATION.
3359            C
3360                     IF(NNQ(ACPOOL).GE.3) THEN
3361                        DO 150 I = 1,NNQ(ACPOOL)
3362                           IF(QSET(LOCAT(I,ACPOOL)+12).EQ.TANKS)NWTANKS=NWTANKS+1
3363              150       CONTINUE
3364                     ENDIF
3365            C
3366            C
3367            C        IF AT LEAST THREE AIRCRAFT AND THREE PILOTS, TRY TO FORM
3368            C        A FLIGHT.
3369            C
3370                     IF(NWTANKS.GE.3.AND.NNQ(PLPOOL).GE.3) THEN
3371            C
3372                        MAX = NNQ(PLPOOL)
3373                        FLTLDS = 2.0
3374                        FORMED =.TRUE.
3375            C
3376            C           DISASSEMBLE THE PILOT READY POOL FOR CLOSER EXAMINATION
3377            C           AND REORGANIZATION (IF NECESSARY).
3378            C
```

```
 3379                      DO 200 I = MAX,1,-1
3380                        CALL ULINK(I,PLPOOL)
3381                        POINT(I) = MFA
3382            200       CONTINUE
3383           C
3384           C          GET FIRST CREW MEMBER - PILOT OF A/C 1
3385           C
3386                      CREW(1) = 0
3387                      FIRST   = 0
3388           C
3389                      I = 1
3390            210       CONTINUE
3391                        IF(QSET(POINT(I)+3).GE.1) THEN
3392                          IF(QSET(POINT(I)+3+DAY).LT.3) THEN
3393                            CREW(1) = I
3394                          ELSE
3395                            IF(FIRST.EQ.0) FIRST = I
3396                          ENDIF
3397                        ENDIF
3398                        I = I + 1
3399                      IF(I.LE.MAX.AND.CREW(1).EQ.0) GO TO 210
3400           C
3401                      IF(CREW(1).EQ.0) CREW(1) = FIRST
3402                      IF(CREW(1).EQ.0) FORMED  = .FALSE.
3403           C
3404           C          FIND SECOND CREW MEMBER - PILOT OF A/C 3
3405           C
3406                      IF(FORMED) THEN
3407                        CREW(3) = 0
3408                        FIRST   = 0
3409                        PILOT   = 0
3410           C
3411                        I = 1
3412            230         CONTINUE
3413                      IF(I.NE.CREW(1))THEN
3414                           IF(QSET(POINT(I))+3.GE.1) THEN
3415                             IF(QSET(POINT(I)+3+DAY).LT.3) THEN
3416                               CREW(3) = I
3417                             ELSE
3418                               IF(FIRST.EQ.0) FIRST = I
3419                             ENDIF
3420                           ELSE
3421                             IF(PILOT.EQ.0) PILOT = I
3422                           ENDIF
3423                           ENDIF
3424                           I = I + 1
3425                        IF(I.LE.MAX.AND.CREW(3).EQ.0) GO TO 230
3426           C
3427                        IF(CREW(3).EQ.0) CREW(3) = FIRST
3428                        IF(CREW(3).EQ.0) THEN
```

```
 3429                          CREW(3) = PILOT
3430                         FLTLDS = 1.0
3431                       ENDIF
3432                       IF(CREW(3).EQ.0)FORMED = .FALSE.
3433                     ENDIF
3434          C
3435          C          GET THIRD CREW MEMBER - PILOT OF A/C Z
3436          C
3437                     IF(FORMED) THEN
3438          C
3439                       CREW(2) = 0
3440                       FIRST   = 0
3441                       FLTLD   = 0
3442                       THREE   = 0
3443          C
3444                       I = 1
3445           220         CONTINUE
3446                         IF(CREW(I).NE.I.AND.CREW(3).NE.I) THEN
3447          C
3448                           IF(QSET(POINT(I)+3).EQ.0) THEN
3449          C
3450                             IF(QSET(POINT(I)+3+DAY).LT.3) THEN
3451                               CREW(2) = I
3452                             ELSE
3453                               IF(FIRST.EQ.0) FIRST = I
3454                             ENDIF
3455          C
3456                           ELSE
3457                             IF(QSET(POINT(I)+3+DAY).LT.3) THEN
3458                               IF(FLTLD.EQ.0)FLTLD=I
3459                             ELSE
3460                               IF(THREE.EQ.0)THREE = I
3461                             ENDIF
3462          C
3463                           ENDIF
3464          C
3465                         ENDIF
3466                         I = I + 1
3467                       IF(I.LE.MAX.AND.CREW(2).EQ.0) GO TO 220
3468          C
3469                       IF(CREW(2).EQ.0) CREW(2) = FIRST
3470                       IF(CREW(2).EQ.0' 'HEN
3471                         CREW(2) = FLTLD
3472                         FLTLDS = 3.0
3473                       ENDIF
3474                       IF(CREW(2).EQ.0) THEN
3475                         CREW(2) = THREE
3476                         FLTLDS = 3.0
3477                       ENDIF
3478                       IF(CREW(2).EQ.0)FORMED=.FALSE.
```

```
 3479                      ENDIF
3480           C
3481           C
3482           C          REASSEMBLE THE PILOT READY POOL
3483           C
3484                      IF(.NOT.FORMED)THEN
3485                        CREW(1) = 1
3486                        CREW(2) = 2
3487                        CREW(3) = 3
3488                      ENDIF
3489           C
3490                      DO 300 I = 1,3
3491                        MFA = POINT(CREW(I))
3492                        CALL LINK(PLPOOL)
3493            300       CONTINUE
3494                      DO 310 I = 1,MAX
3495                        IF(I.NE.CREW(1).AND.I.NE.CREW(2).AND.I.NE.CREW(3))THEN
3496                          MFA = POINT(I)
3497                          CALL LINK(PLPOOL)
3498                        ENDIF
3499            310       CONTINUE
3500           C
3501                    ENDIF
3502           C
3503           C
3504                    J = J + 1
3505                  IF(J.LE.6.AND..NOT.FORMED)GOTO 100
3506           C
3507           C
3508           C     RESET THE PRIORITY SEQUENCE
3509           C
3510                  IF(FORMED.AND.J.LE.6)THEN
3511                    DO 400 I = J,6
3512                      PRIORTY(I - 1) = PRIORTY(I)
3513            400     CONTINUE
3514                    PRIORTY(6) = NSQ
3515                  ENDIF
3516           C
3517           C
3518                  RETURN
3519                  END
3520           C
3521           C
3522           C
3523           C
3524           C
```

```
      SUBROUTINE NIGHTPK(NCALLS,NPARK,NTYPE)
C
      COMMON QSET(1)
      COMMON/PFLAG/MAXPRT,LEVPRT,BPRT,EPRT
      DIMENSION NSET(1),NPARK(6,50,2),NTYPE(6,3)
      EQUIVALENCE (NSET(1),QSET(1))
C
C
C     PERFORM THE NIGHT PARKING ACTIVITY.  DOUBLE UP AIRCRAFT
C     IN THE SHELTERS TILL MORNING (EXCEPT THE QRA SHELTERS).
C
      DO 100 I = 1,6
      IF(NNQ(I*3-2).NE.0) THEN
C
         NBEGIN = 1
         IF(NCALLS.GE.2) NBEGIN = NNQ(I*3-2)
         DO 200 J = NBEGIN,NNQ(I*3-2)
            LOC = LOCAT(J,I*3-2)
            NSQ = QSET(LOC+1)
            IF(QSET(LOC+3).NE.1) THEN
               K = 0
 220           IF(K.GE.NTYPE(NSQ,1)) GO TO 260
               K = K + 1
               IF(NPARK(NSQ,K,2).NE.0) GO TO 220
C
               NPARK(NSQ,K,2) = QSET(LOC+2)
               QSET(LOC+3) = 1
               K = 6
 240           IF(K.GE.50) GO TO 260
               K = K + 1
               IF(NPARK(NSQ,K,1).NE.QSET(LOC+2)) GO TO 240
               NPARK(NSQ,K,1) = 0
 260           CONTINUE
            ENDIF
 200     CONTINUE
C
      ENDIF
 100  CONTINUE
C
      RETURN
      END
C
C
C
C
C
```

```
3571                SUBROUTINE QRASWCH(NCSQ,NPARK)
3572               COMMON QSET(1)
3573               COMMON/PFLAG/MAXPRT,LEVPRT,BPRT,EPRT
3574               INTEGER NCSQ(300),NSET(1),NPARK(6,50,2),MARK(75)
3575               EQUIVALENCE (NSET(1),QSET(1))
3576         C
3577         C
3578         C     CHANGE OVER THE QRA AIRCRAFT OF THE RESUPPLIED SQUADRONS
3579         C
3580               DO 100 I = NNQ(19),1,-1
3581                  NSQ = QSET(LOCAT(I,19)+1)
3582                  NTAIL = QSET(LOCAT(I,19)+2)
3583                  IF(NNQ(NSQ*3-2).GT.0.AND.NSQ.NE.NCSQ(NTAIL))THEN
3584                     CALL ULINK(I,19)
3585                     CALL LINK(NCSQ(NTAIL)*3-2)
3586                     JTAIL = QSET(LOCAT(NNQ(NSQ*3-2),NSQ*3-2)+2)
3587                     CALL ULINK(NNQ(NSQ*3-2),NSQ*3-2)
3588                     CALL LINK(19)
3589                     IF(NPARK(NSQ,1,1).EQ.NTAIL) NPARK(NSQ,1,1) = JTAIL
3590                     IF(NPARK(NSQ,2,1).EQ.NTAIL) NPARK(NSQ,2,1) = JTAIL
3591                     IF(NPARK(NSQ,3,1).EQ.NTAIL) NPARK(NSQ,3,1) = JTAIL
3592                     DO 200 K = 1,50
3593                     DO 200 J = 1,2
3594                        IF(NPARK(NSQ,K,J).EQ.JTAIL) NPARK(NSQ,K,J) = NTAIL
3595          200        CONTINUE
3596                  ENDIF
3597          100  CONTINUE
3598         C
3599         C
3600         C     CHANGE OVER THE QRA PILOTS FOR ALL THE SQUADRONS
3601         C
3602               DO 250 I = 1,6
3603                  K = 0
3604                  DO 300 J = 1,NNQ(20)
3605                     NL = LOCAT(J,20)
3606                     IF(QSET(NL+1).EQ.I)THEN
3607                        K = K + 1
3608                        MARK(K) = NL
3609                     ENDIF
3610          300     CONTINUE
3611         C
3612                  M = 0
3613                  IF(NNQ(I*3-1).GT.0.AND.K.GT.0)THEN
3614                  DO 400 J = NNQ(I*3-1),1,-1
3615                     NEW = LOCAT(J,I*3-1)
3616                     IF(QSET(NEW+3).GE.2.0.AND.M.LT.K)THEN
3617                        QSET(NEW+3) = 3.0
3618                        M = M + 1
3619                        QSET(MARK(M)+3) = 2.0
3620                        CALL ULINK(-MARK(M),20)
```

```
 3621                        CALL LINK(I*3-1)
3622                        CALL ULINK(-NEW,I*3-1)
3623                        CALL LINK(20)
3624                      ENDIF
3625          400     CONTINUE
3626                  ENDIF
3627          250  CONTINUE
3628         C
3629         C
3630              RETURN
3631              END
3632         C
3633         C
3634         C
3635         C
3636         C
```

```
      SUBROUTINE RESUPLY(ACTIVE,NCSQ,LOSTFTL,LIMITAC,REQSUPY,TNOW)
      COMMON QSET(1)
      COMMON/PFLAG/MAXPRT,LEVPRT,BPRT,EPRT
C
      INTEGER NOPER(6),NCSQ(300),LOSTFTL(6)
      INTEGER NSET(1)
      EQUIVALENCE (NSET(1),QSET(1))
      LOGICAL REQSUPY(6),ACTIVE(6)
C
      DO 50 I = 1,6
         NOPER(I) = 0
         REQSUPY(I) = .FALSE.
   50 CONTINUE
C
C     DETERMINE THE TOTAL NUMBER OF AIRCRAFT PER SQUADRON.
C
      DO 60 I = 1,300
      IF(NCSQ(I).GT.0.AND.NCSQ(I).LE.6)THEN
         NSQ = NCSQ(I)
         NOPER(NSQ) = NOPER(NSQ) + 1
      ENDIF
   60 CONTINUE
C
      IF(LEVPRT.GE.3.AND.TNOW.GE.BPRT.AND.TNOW.LE.EPRT)THEN
         PRINT*,'-A/C OPERATIONAL BY SQDN: ',(NOPER(I),I=1,6)
      ENDIF
C
C
C     CHECK TO SEE IF ALL SQUADRONS IN A WING HAVE ENOUGH AIR-
C     CRAFT BETWEEN THEM TO FORM A FLIGHT.
C
      NBEGIN = 1
      NSTOP  = 6
      IF(NOPER(1)+NOPER(2)+NOPER(3).LT.LIMITAC) THEN
         IF(ACTIVE(1)) REQSUPY(1) = .TRUE.
         IF(ACTIVE(2)) REQSUPY(2) = .TRUE.
         IF(ACTIVE(3)) REQSUPY(3) = .TRUE.
         NBEGIN  = 4
      ENDIF
      IF(NOPER(4)+NOPER(5)+NOPER(6).LT.LIMITAC) THEN
         IF(ACTIVE(4)) REQSUPY(4) = .TRUE.
         IF(ACTIVE(5)) REQSUPY(5) = .TRUE.
         IF(ACTIVE(6)) REQSUPY(6) = .TRUE.
         NSTOP = 3
      ENDIF
C
      DO 100 I = NBEGIN,NSTOP
C
      IF(LEVPRT.GE.3.AND.TNOW.GE.BPRT.AND.TNOW.LE.EPRT)THEN
         PRINT*,'-PROCESSING SQDN ',I
```

```
 3687                ENDIF
3688          C
3689              IF(NOPER(I).LT.LIMITAC.AND.ACTIVE(I))THEN
3690          C
3691                IF(LEVPRT.GE.3.AND.TNOW.GE.BPRT.AND.TNOW.LE.EPRT)THEN
3692                  PRINT*,'-RESUPPLY SQDN ',I
3693                ENDIF
3694          C
3695                REQSUPY(I) = .TRUE.
3696          C
3697          C     DETERMINE THE SISTER SQUADRONS OF SQUADRON I
3698          C
3699                NSQ1 = 1 + (I/4)*3
3700                IF(I.EQ.1.OR.I.EQ.4) NSQ1 = 2 + (I/4)*3
3701                NSQ2 = 3 + (I/4)*3
3702                IF(I.EQ.3.OR.I.EQ.6) NSQ2 = 2 + (I/4)*3
3703          C     CALCULATE THE NUMBER OF AIRCRAFT THE SISTER SQUADRONS
3704          C     REQUIRE TO BECOME OPERATIONAL (NOPER.GE.LIMITAC) IF THE
3705          C     SISTER SQUADRON IS NOT BEING RESUPPLIED.
3706          C
3707                NREQ1 = MAX(0,LIMITAC-NOPER(NSQ1))
3708                NREQ2 = MAX(0,LIMITAC-NOPER(NSQ2))
3709          C
3710                NCNT  = 0
3711                MOVE1 = 0
3712                MOVE2 = 0
3713          C
3714          C
3715          C     IF A SQUADRON REQUIRES RESUPPLY, ITS ARCRAFT ARE FARMED
3716          C     OUT TO ITS SISTER SQUADRONS ACCORDING TO THE FOLLOWING
3717          C     RULES (UNLESS ONE OF THE SISTER SQUADRONS IS ALSO BEING
3718          C     RESUPPLIED, IN WHICH CASE, ALL THE AIRCRAFT ARE FARMED
3719          C     TO THE OTHER SISTER SQUADRON):
3720          C
3721          C     1.  IF THE SECOND SISTER SQUADRON IS NOT OPERATIONAL
3722          C         (NOPER < LIMITAC), THEN IT RECEIVES AIRCRAFT UNTIL
3723          C         IT IS AT OPERATIONAL STRENGTH.
3724          C     2.  THE FIRST SISTER SQUADRON IS THEN BROUGHT UP TO
3725          C         STRENGTH.
3726          C     3.  THE TWO SISTER SQUADRONS ARE BROUGHT UP TO EQUALL
3727          C         NUMBERS, AS LONG AS THEIR ARE STILL AIRCRAFT.
3728          C     4.  FINALLY, IF ANY AIRCRAFT ARE LEFT, THEY ARE DISTRIB-
3729          C         UTED EQUALLY BETWEEN THE TWO SISTER SQUADRONS.
3730          C
3731                DO 200 J = 1,300
3732                IF(NCSQ(J).EQ.I) THEN
3733          C
3734                  IF(REQSUPY(NSQ1))THEN
3735                    NEXT = NSQ2
3736                  ELSEIF(REQSUPY(NSQ2))THEN
```

```
                        NEXT = NSQ1
                     ELSEIF(MOVE2.LT.NREQ2)THEN
                        NEXT = NSQ2
                     ELSEIF(MOVE1.LT.NREQ1)THEN
                        NEXT = NSQ1
                     ELSEIF(NOPER(NSQ1).LT.NOPER(NSQ2))THEN
                        NEXT = NSQ1
                     ELSE
                        NEXT = NSQ2
                     ENDIF
C
                     NCSQ(J) = NEXT
                     NOPER(NEXT) = NOPER(NEXT) + 1
                     NOPER(I) = NOPER(I) - 1
                     IF(NEXT.EQ.NSQ1)MOVE1 = MOVE1 + 1
                     IF(NEXT.EQ.NSQ2)MOVE2 = MOVE2 + 1
C
                  ENDIF
 200              CONTINUE
C
                  IF(LEVPRT.GE.3.AND.TNOW.GE.BPRT.AND.TNOW.LE.EPRT)THEN
                  PRINT*,'-A/C AFTER RESUPPY OF ',I,' IS ',(NOPER(J),J=1,6)
                  ENDIF
C
C        MOVE A/C FROM ONE POOL TO ANOTHER
C
                  DO 300 J = NN2(I*3-2),1,-1
C
                     NTAIL = QSET(LOCAT(J,I*3-2)+2)
                     CALL ULINK(J,I*3-2)
                     IF(NCSQ(NTAIL).EQ.NSQ1) THEN
                        CALL LINK(NSQ1*3-2)
                     ELSE
                        CALL LINK(NSQ2*3-2)
                     ENDIF
 300              CONTINUE
C
C
C     NEXT, FARM OUT THE PILOTS.  EACH SISTER SQUADRON GETS ITS
C     FLIGHT LEADS REPLACED BY SQUADRON I (AS LONG AS SQUADRON
CC    I HAS SOME).  AFTER THE FLIGHT LEADS ARE FARMED OUT, EACH
C     SISTER SQUADRON RECEIVES AS MANY TOTAL PILOTS FROM SQUAD-
C     RON I AS THEY RECEIVEC AIRCRAFT.  THE REMAINING PILOTS
C     BECOME PART OF THE REPLACEMENT SQUADRON.
C
                     NPL1 = LOSTFTL(NSQ1)
                     NPL2 = LOSTFTL(NSQ2)
C
                     IF(NPL2.GT.NPL1) THEN
C
```

```
3787                    LEN = MIN(NPL2-NPL1,MOVE2)
3788                  J = NNQ(I*3-1)
3789          400     IF(J.LE.0.OR.LEN.EQ.0) GO TO 500
3790                    IF(QSET(LOCAT(J,I*3-1)+3).GE.1) THEN
3791                      QSET(LOCAT(J,I*3-1)+1) = NSQ2
3792                      CALL ULINK(J,I*3-1)
3793                      CALL LINK(NSQ2*3-1)
3794                      MOVE2 = MOVE2 - 1
3795                      LEN   = LEN   - 1
3796                    ENDIF
3797                    J = J - 1
3798                  GO TO 400
3799          500     CONTINUE
3800          C
3801                ELSEIF(NPL1.GT.NPL2) THEN
3802          C
3803                  LEN = MIN(NPL1 -NPL2,MOVE1)
3804                  J =NNQ(I*3-1)
3805          600     IF(J.LE.0.OR.LEN.EQ.0) GO TO 700
3806                    IF(QSET(LOCAT(J,I*3-1)+3).GE.1) THEN
3807                      QSET(LOCAT(J,I*3-1)+1) = NSQ1
3808                      CALL ULINK(J,I*3-1)
3809                      CALL LINK(NSQ1*3-1)
3810                      MOVE1 = MOVE1 - 1
3811                      LEN   = LEN   - 1
3812                    ENDIF
3813                    J = J - 1
3814                  GO TO 600
3815          700     CONTINUE
3816          C
3817                ENDIF
3818          C
3819                J = NNQ(I*3-1)
3820          800   IF(J.LE.0.OR.MOVE2.LE.0)GOTO 900
3821                  MOVE2 = MOVE2 - 1
3822                  QSET(LOCAT(J,I*3-1)+1) = NSQ2
3823                  CALL  ULINK(J,I*3-1)
3824                  CALL LINK(NSQ2*3-1)
3825                  J = J - 1
3826                GOTO 800
3827          900   CONTINUE
3828          C
3829                J = NNQ(I*3-1)
3830          1000  IF(J.LE.0.OR.MOVE1.LE.0)GOTO 1100
3831                  MOVE1 = MOVE1 - 1
3832                  QSET(LOCAT(J,I*3-1)+1) = NSQ1
3833                  CALL ULINK(J,I*3-1)
3834                  CALL LINK(NSQ1*3-1)
3835                  J = J - 1
3836                GOTO 1000
```

```
 3837              1100    CONTINUE
3838            C
3839                  ENDIF
3840               100  CONTINUE
3841            C
3842                  RETURN
3843                  END
3844            C
3845            C
3846            C
3847            C
```

```
      SUBROUTINE RECONFG(INITAC,MAXCONF,NACTYPE,NAVAIL,TNOW)
      COMMON QSET(1)
      COMMON/PFLAG/MAXPRT,LEVPRT,BPRT,EPRT
C
      DIMENSION NSET(1),NCUM(7),NWANT(3),INITAC(3),INITSQ(3),MAXCONF(6)
     &,NACTYPE(6,3),NAVAIL(6)
      EQUIVALENCE (NSET(1),QSET(1))
C
C     AIRCRAFT ARE RECONFIGURED FOR THE NEXT DAY ACCORDING TO THE
C     INITIAL CONFIGURATION RATIOS, INITAC, SET BY THE USER.
C     ADDITIONALLY, ALL THE AIRCRAFT IN A PARTICULAR SQUADRON ARE
C     CONFIGURED THE SAME.
C
C
C     DETERMINE THE NUMBER OF AIRCRAFT PER CONFIGURATION.
C
      NCUM(1) = 0
      DO 100 I = 2,7
         NCUM(I) = NCUM(I-1) + NAVAIL(I-1)
  100 CONTINUE
      INITRDY = INITAC(1) + INITAC(2) + INITAC(3)
      DO 200 I = 1,3
         NWANT(I) = FLOAT(INITAC(I))/FLOAT(INITRDY)*FLOAT(NCUM(7))+0.5
         INITSQ(I) = 0
  200 CONTINUE
C
      IF(LEVPRT.GE.3.AND.TNOW.GE.BPRT.AND.TNOW.LE.EPRT)THEN
         PRINT*,'-NUMBER OF A/C WANTED PER CONFIGURATION (3,2,1): '
         PRINT*,'          ',(NWANT(I),I=3,1,-1)
      ENDIF
C
C
C     DETERMINE THE NUMBER OF SQUADRONS PER CONFIGURATION.
C
      K = 1
  300 IF(NWANT(3).LE.NCUM(K)) GO TO 400
         K = K + 1
         INITSQ(3) = INITSQ(3) + 1
         IF(K.LE.7) GO TO 300
  400 IF(K.LE.7.AND.NWANT(3)+NWANT(2).LE.NCUM(K)) GO TO 500
         K = K + 1
         INITSQ(2) = INITSQ(2) + 1
         IF(K.LE.7) GO TO 400
  500 INITSQ(1) = 6 - INITSQ(3) - INITSQ(2)
C
      IF(LEVPRT.GE.3.AND.TNOW.GE.BPRT.AND.TNOW.LE.EPRT)THEN
         PRINT*,'-NUMBER OF SQDN REQ PER CONFIGURATION (3,2,1): '
         PRINT*,'          ',INITSQ(3),INITSQ(2),INITSQ(1)
      ENDIF
C
```

```
C
C     RECONFIGURE THE AIRCRAFT.
C
      DO 600 I = 1,6
         MAXCONF(I) = 1
         IF(I.LE.INITSQ(2)+INITSQ(3)) MAXCONF(I) = 2
         IF(I.LE.INITSQ(3)) MAXCONF(I) = 3
         DO 620 J = 1,3
            NACTYPE(I,J) = 0
 620     CONTINUE
         NACTYPE(I,MAXCONF(I)) = NNQ(I*3-2)
         DO 640 J = 1,NNQ(I*3-2)
            QSET(LOCAT(J,I*3-2)+12) = MAXCONF(I)
 640     CONTINUE
 600  CONTINUE
C
      IF(LEVPRT.GE.3.AND.TNOW.GE.BPRT.AND.TNOW.LE.EPRT)THEN
         PRINT*,'-MAXIMUM CONFIGURATION BY SQUADRON: '
         PRINT*,'        ',(MAXCONF(I),I=1,6)
      ENDIF
C
      RETURN
      END
```

Appendix C: Notes to Users

Introduction

This appendix has been included to assist a user in providing values for several of the more complicated variables in Subroutines INTLC and USERI. There is no intention in these notes to include information presented elsewhere in chapters I through IV, and Appendix A and B. Rather, this guide will expand on previously covered material and go into greater depth where required to assist the user in specifying:

(1) Fragmentary order variables
(2) Mission variables
(3) Maintenance failure variables
(4) Crash/Tow decision matrices
(5) Airfield facility composition/definition
(6) Settings for the Master Clock

Of necessity, some variables are less well defined than others in the comments in Subroutines INTLC and USERI. These notes will clarify those variables when used in conjunction with Annex A. These notes conclude with

comments on computer run time and a sample CDC 6600 day file. All line number references in this guide are to Appendix B unless otherwise stated.

The Fragmentary Order

The frag order requires four variables be defined. The variables are inputs to the Scheduler (Event 3). Sortie rates are input to Data Statement SRATE, by day. Refer to Appendix B, page 131 for all variables in this discussion.

Gaggles are only flown to Area 3. If gaggles are desired to Area 2 specify gaggles and give all the Area 3 mission variables the same values and probabilities as Area 2. The percentage of sorties desired to each area is specified in Data Statement REQPCT. These values give the Scheduler a target to shoot for.

With the sortie rates, area percentages, and size of gaggles desired, the remainder of the information is straightforward. Data Statement NGAGLE is specified using the number of flights desired in up to two gaggles per day. As an example, lines 64 and 66 show 8 and 8 for Day 1 only. This means two gaggles of 8 three-ships are desired on Day 1 only. Normally, a gaggle launch requires a special effort in the real world and there is a delay time before sorties

begin launching again in a steady flow. This is specified in Data Statement DELAY, lines 71 and 73 (zero delay may be specified). The values are in minutes. If no gaggles are desired, input zeros in NGAGLE.

Data Statement INITAC also has a left-handed influence on the Scheduler. Aircraft are configured in the night routines based on the numbers in INITAC. Using the input values, the coding decides how many squadrons to configure for areas 3, 2, and 1, in order. A clever user can approach simulating maintenance turnaround scheduling by playing with the numbers. If both gaggles are desired back to back in the morning, then enough Area 3 configured aircraft must be specified in INITAC to meet the requirement. Otherwise turnaround comes into play and the simulation deals with maintenance type scheduling.

Mission Variables

Duration times are at lines 104 to 106 and are triangular (minimum, most likely, maximum). These values are straightforward, as are attrition rates, lines 113 to 118. Sensitivity analysis should be reviewed for attrition rates. Similarly, tank jettison is at lines 121 to 125. Ordnance probabilities are at lines 462 to 480. They are

aggregated over the entire theater, and a macro viewpoint must be adopted to set them. For example, PBOMCP is one aggregated value. When set at 1.0, it is assumed an attempt will always be made to drop or jettison. Then, the only way aircraft return with bombs is due to a malfunction. The gun can only malfunction if it is used. Battle damage and damage level probabilities are at line 484 to 493.

Maintenance Failures

Introduction. Maintenance failures occur when ATOF exceeds the current value of total engine running time. This will occur at an update point following an activity where the aircraft engine was running, as previously described in Chapter III.

Level of Failure. To determine the level of failure the coding uses the cumulative probabilities in lines 515 to 526.

Data Statement SYSTOL. A system may also fail if an aircraft is delayed at preflight. Data Statement SYSTOL allows the user to specify the number of minutes away from failure a system can be when it is considered broken at preflight. For example, the 5.0 values in line 533 mean that if a system has equal to or less than 5.0 minutes to

failure, it will fail at preflight delay.

Data Statement NBATREP. When aircraft incur battle damage (Attribute(16)) on a mission and subsequently return to base, a determination is made as to whether they are repairable (node BADA, Appendix A, page 102). Level four and five battle damage were conceptually considered unrepairable at the local level. This is shown in line 544 by the 999999s in the fourth and fifth places of NBATREP. LEVEL 1, 2, and 3 damage is converted to a user specified equivalent maintenance failure code found in NBATREP. After the equivalent code is set, it is combined with the current aircraft failure code. This is done digit by digit, using the highest value of either code in each digit. The aircraft then is processed into maintenance. Aircraft with a 999999 value are conceptually considered to be awaiting depot level repair when available. The 999999 aircraft are available for cannibalization even if they are total losses. The 999999 aircraft are routed to the JUNK file.

Service Times. Service times must also be set for each section of maintenance. Wing and squadron service times are set in lines 361 to 387. AMT service times are specified between lines 402 and 413. Between lines 391 and 396 are interference times which are added to wing service when a single shop is repairing two systems concurrently as can happen in shops 1 and 3. Between lines 416 and 422

interference times are specified for squadron repairs. These are added for interference in concurrent service.

Mean Time between Failure (MTBF). The values for MTBF are input in Subroutine USERI, at line 130. These values should be as carefully chosen as possible, and sensitivity analysis should be performed on this set of variables.

Beta Distribution Shape Parameters. The use of the shape parameters in calculating MTOF was covered in Chapter III. The process for determining the beta distribution shape parameters consists of four steps.

(1) The user must form an opinion of the reliability of each system.

(2) The user should study histograms, or curves, of the Beta distribution with various parameters, to select a set of parameters for each system. The parameters should reflect the user's view of the reliability of the particular system.

(3) The user should generate a series of histograms, or curves, around these parameters to insure the shapes are really what is desired, and to insure the shapes reflect the user's opinion of the relative reliability of the systems.

(4) The user should use the parameters selected for each system as inputs to Data Statements ALP and

BET at lines 132 and 134.

Each of the steps is relatively self-explanatory, however a short recap is probably useful. Forming an opinion of reliablity requires that opinions be made explicit -- not implicit. The shape of the curve will identify the biases. Biases are fine -- as long as they are explicit and sensitivity analysis may be performed.

Using the SLAM program in Figure C.1, sets of sample histograms were generated. In this case, histograms for the 10 SLAM random number streams were generated for the three ALPHA and BETA pairs (5.0,1.5), (3.0,1.5), and (4.0,2.0). This is the family of parameters used in the analysis in this document. The results are presented in Figure C.2.

This family of shape parameters represented the authors' beliefs in the systems' reliabilities. The family of parameters boiled down to three cases. Case I was most reliable (ALPHA = 5.0, BETA = 1.5). The curve is shaped skewed, or humped, to the right. Most failures occur in a range around or near the MTBF. See figures C.2.1 and C.2.2.

Case II is slightly less reliable (ALPHA = 4.0, BETA = 2.0). The right skew is not so pronounced. See Figures C.2.3 and C.2.4. Case III is least reliable (ALPHA = 3.0, BETA = 1.5). The shape is tending toward normality. See Figures C.2.5 and C.2.6. Refer to Annex A for further

```
.*  SLAM STATEMENTS USED TO GENERATE THE BETA HISTOGRAMS.
.*
.*
RWM,CM150000,T100,IO200.  T000845,MANN,BOX4566,AFIT,AFIT,AFIT,91,91,91
ATTACH,PROCFIL,ID=A810171,SN=ASDAD.
BEGIN,NOSFILE.
GET,BETABIN,ID=COVEY.
REWIND,BETABIN.
ATTACH,PROCFIL,SLAMPROC,ID=AFIT.
BEGIN,SLAM,,M=BETABIN,PL=100000.
*EOR
GEN,SLMBETA,MANN & SHOOK 91 AFIT,1/8/82,3,YES,NO,YES,NO,NO;
LIMITS,0,1,100;
NETWORK;
      CREATE,.1,,,1000,1;
      ASSIGN,ATRIB(1) = USERF(1);
      TERMINATE;
      ENDNETWORK;
INIT,0,500;
INTLC,XX(1)=5,XX(2)=1.5;
SIMULATE;
INTLC,XX(1)=3,XX(2)=1.5;
SIMULATE;
INTLC,XX(1)=4,XX(2)=2;
FIN;



C     THE FOLLOWING FORTRAN SUPPLIMENTS THE ABOVE SLAM TO PRODUCE
C     THE HISTOGRAMS OF THE BETA DISTRIBUTIONS.


      SUBROUTINE INTLC
      COMMON/D/NCNT,DAT(10,1000)
      NCNT = 0
      RETURN
      END


      FUNCTION USERF(IFN)
      COMMON/SCOM1/ATRIB(100),DD(100),DDL(100),DTNOW,II,MFA,MSTOP,NCLNR
     &,NCRDR,NPRNT,NNRUN,NNSET,NTAPE,SS(100),SSL(100),TNEXT,TNOW,XX(100)
      COMMON/D/NCNT,DAT(10,1000)
      USERF = 0.0
      NCNT = NCNT + 1
      DO 100 I = 1,10
         DAT(I,NCNT) = BETA(XX(1),XX(2),I)
  100 CONTINUE
      RETURN
      END
```

Fig. C.1.1 Beta Histogram Generation Program

```
      SUBROUTINE OTPUT
      COMMON/SCOM1/ATRIB(100),DD(100),DDL(100),DTNOW,II,MFA,MSTOP,NCLNR
     &,NCRDR,NPRNT,NNRUN,NNSET,NTAPE,SS(100),SSL(100),TNEXT,TNOW,XX(100)
      COMMON/D/NCNT,DAT(10,1000)
      INTEGER NUM(21)
      CHARACTER OUTL(50)

        DO 10 K = 1,10
         PRINT 11,K,XX(1),XX(2)
11       FORMAT('1','       GRAPH OF STREAM ',I2,'    ALPHA ',F4.1,
     &  '    BETA ',F4.1)
         PRINT*,
         PRINT*,
         DO 50 I = 1,21
            NUM(I) = 0
50       CONTINUE
         DO 100 I = 1,1000
            INDEX = DAT(K,I)*20 + 1
            IF(INDEX.GT.21)INDEX = 21
            NUM(INDEX) = NUM(INDEX) + 1
100      CONTINUE
         DO 150 I = 1,21
            LINE = NUM(I)/4
            DO 200 J = 1,50
               OUTL(J) = ' '
200         CONTINUE
            IF(LINE.GT.50)THEN
               LINE = 49
               OUTL(50) = '+'
            ENDIF
            DO 250 J = 1,LINE
               OUTL(J) = '*'
250         CONTINUE
            PRINT 251,NUM(I),(OUTL(L),L=1,50)
251         FORMAT(' ',10X,'(',I4,')',3X,'I ',50A1)
150      CONTINUE

           DO 300 J = 1,50
            OUTL(J) = '-'
300      CONTINUE
         PRINT 302,(OUTL(J),J=1,50)
302      FORMAT(' ',19X,'+ ',50A1)
         PRINT 301
301      FORMAT(' ',21X,'4',47X,'200')
10    CONTINUE

        RETURN
      END
```

Fig. C.1.2 Beta Histogram Generation Program

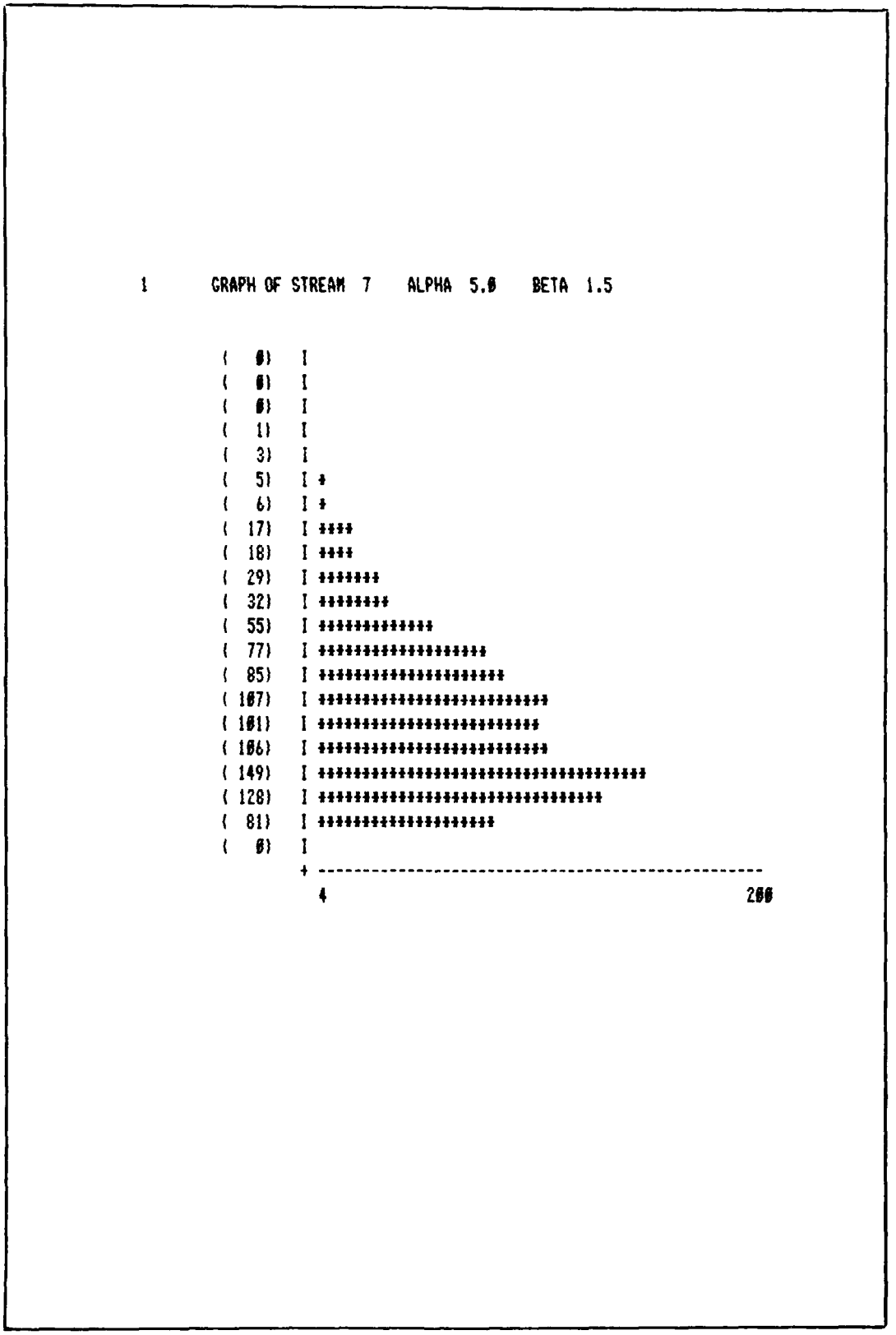


Fig. C.2.1 Case I Histogram

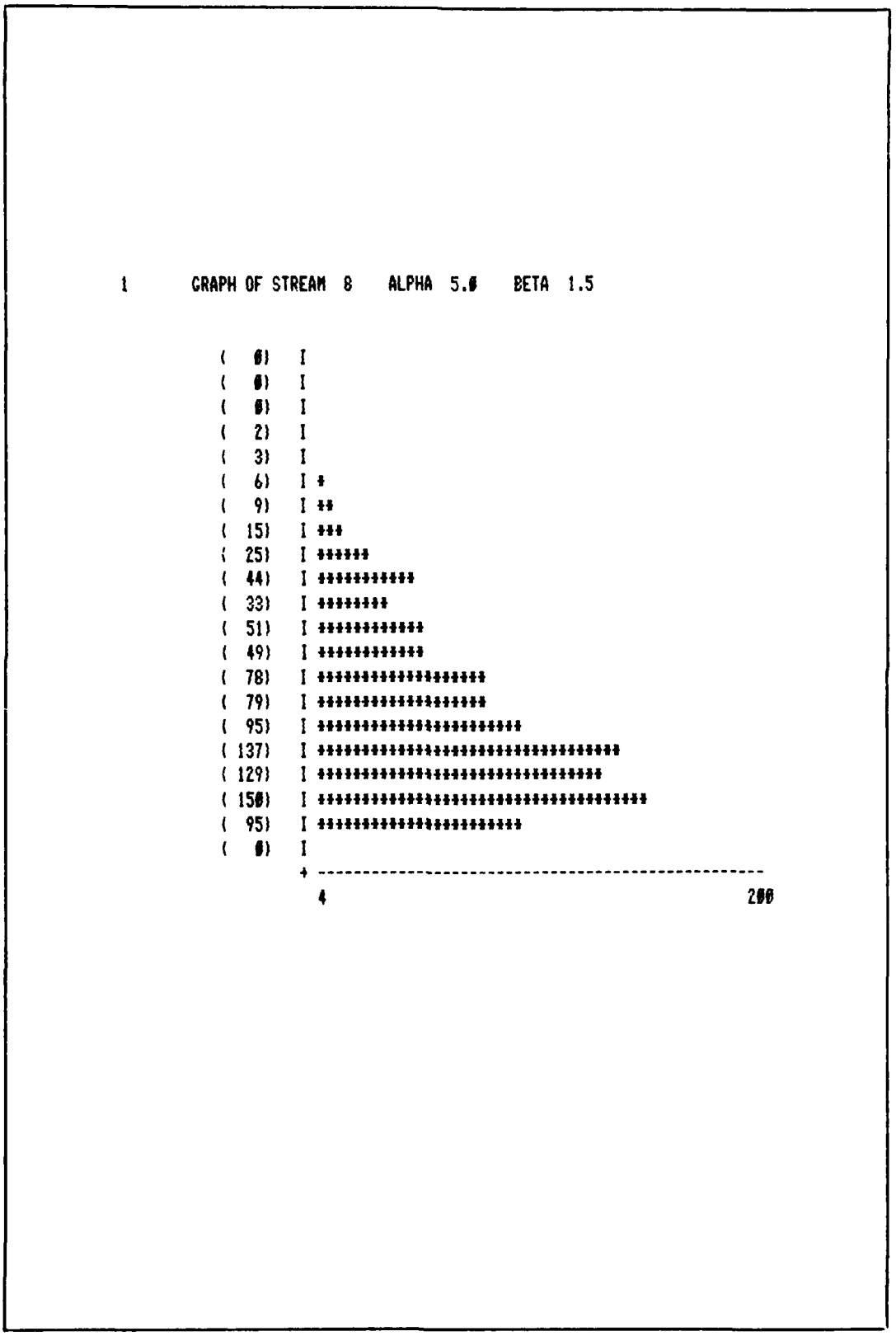


Fig. C.2.2 Case I Histogram

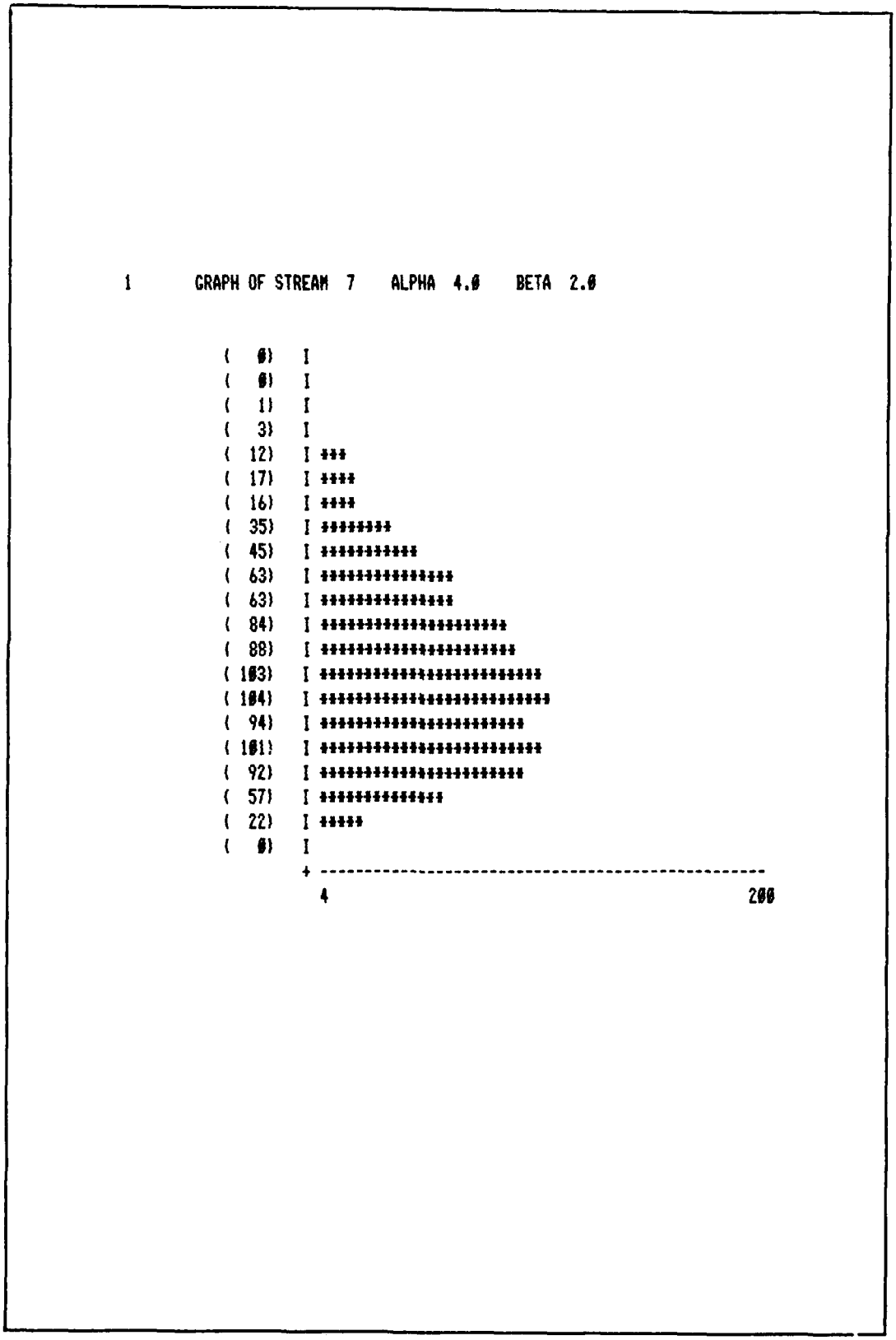


Fig. C.2.3 Case II Histogram

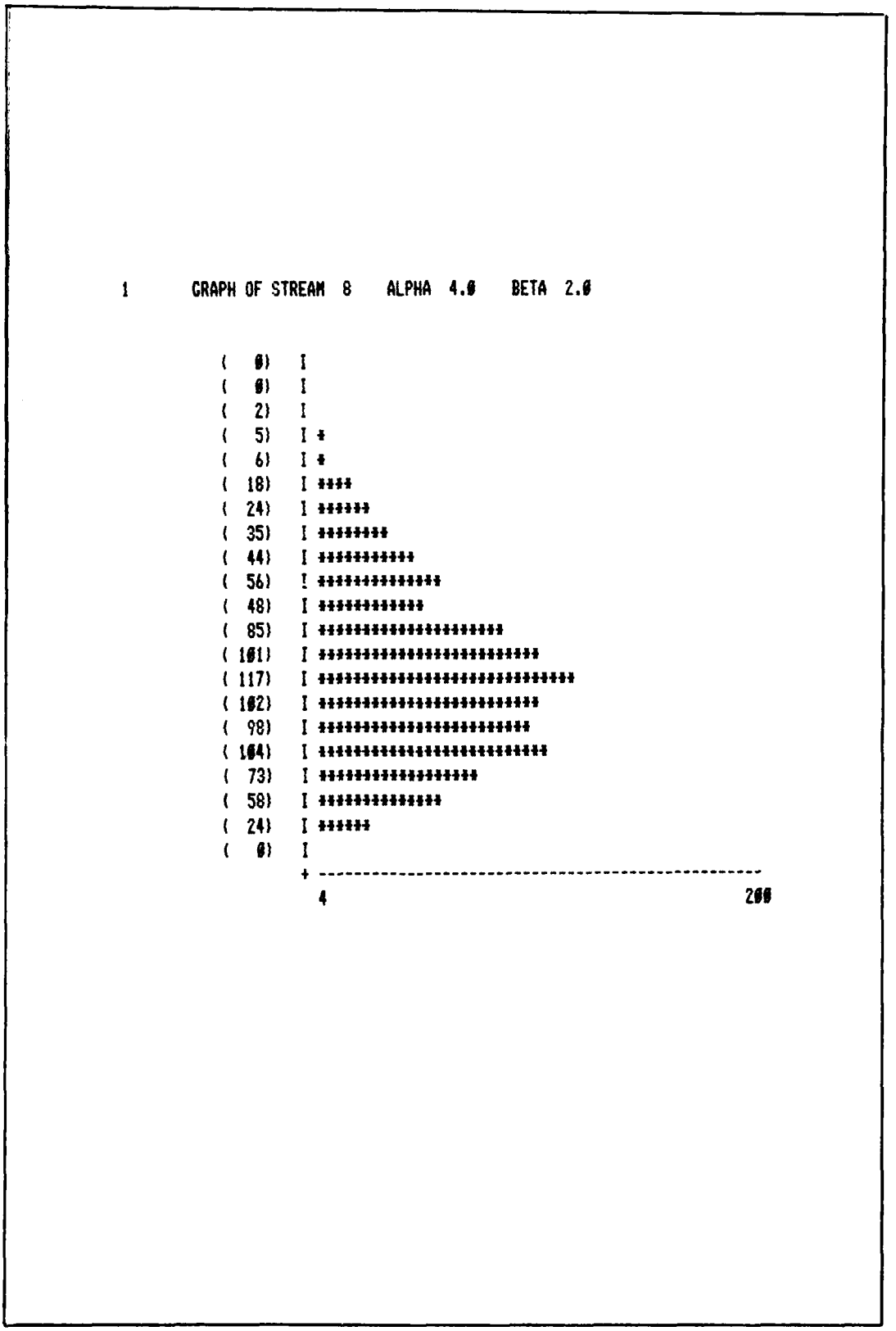


Fig. C.2.4 Case II Histogram

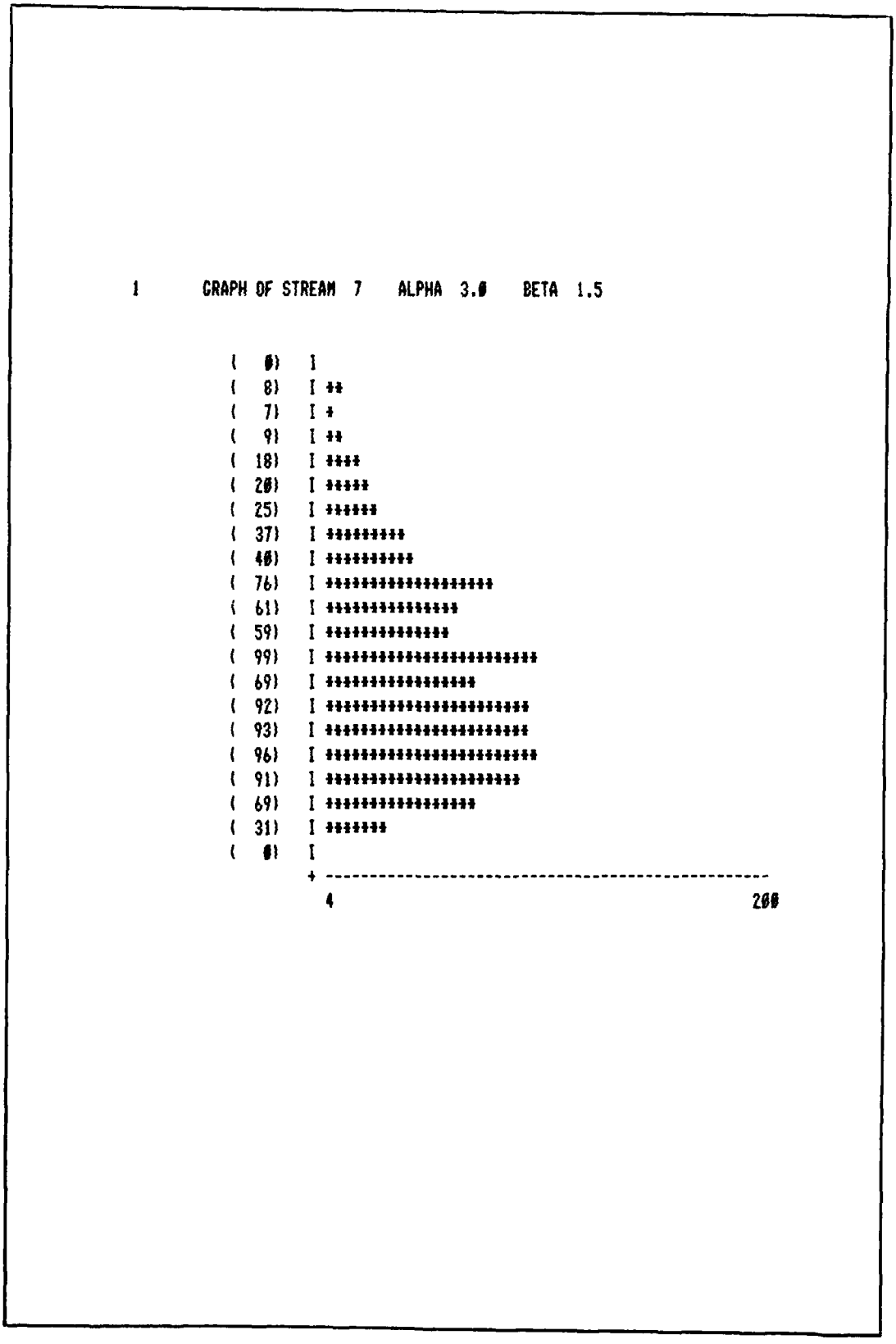


Fig. C.2.5 Case III Histogram

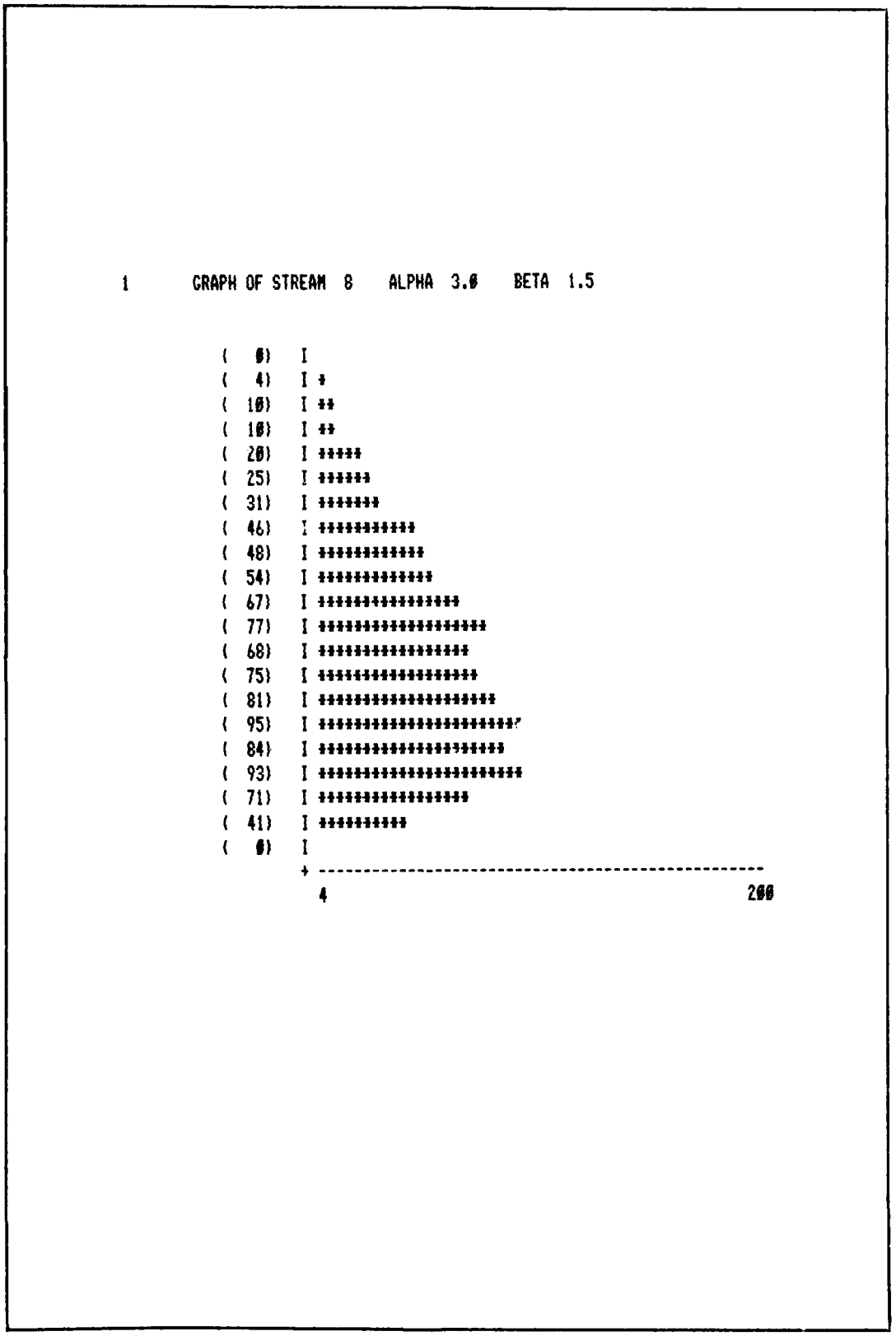


Fig. C.2.6 Case III Histogram

comments.

Once tentative shape parameters have been selected, a family of histograms should be generated and studied to insure that both absolute and relative reliability are being reflected as desired. Finally, input the parameters as explained above.

Crash/Tow Decision Matrices

Introduction. When aircraft malfunctions occur, routing decisions must be made. While airborne, aircraft may crash due to maintenance failure, or battle damage, or a combination of both. On the ground aircraft may require towing. The variables (LCRSH, LTOW, LBAT) used to determine the state of the aircraft after a malfunction are contained between lines 430 and 458. Each matrix (vector) can contain up to 24 encoded numbers. Each encoded number represents a combination of system failure levels.

LCRSH. When a malfunction occurs in the air, the network uses the appropriate FORTRAN routine to compare the aircraft's failure code to the appropriate matrix, digit by digit (system by system). If an aircraft system level is greater than or equal to the corresponding encoded number digit for all systems, the aircraft will crash. The

aircraft is compared to every encoded number, up to the point its status is determined.

Set the values in NCRSH/NTOW/NBAT equal to the number of encoded numbers in each matrix (vector). Refer to the comments between lines 430 and 447.

LTOW. This vector is used in a similar manner to LCRSH. The only difference is that the levels specified in the encoded numbers represent combinations of system failure levels which cause an aircraft to require towing. For example, if an aircraft will require towing when the level of systems 1 to 4 is greater than or equal to 3, then the encoded number would be included as 333300. All reasonable combinations should be included. Set NCRSH/NTOW/NBAT accordingly in line 443.

LBAT. This vector is similar to the above, except that it also considers the influence of battle damage on the possibility of crashing. This is done by adding a seventh digit to the encoded numbers. The battle damage digit is the most significant (first digit), and the next six are aircraft systems of the aircraft failure code. The numbers are encoded as in LCRSH and stored in the same fashion except an additional 9 is added for the crash code. Set NCRSH/NTOW/NBAT (line 443) according to the number of encoded numbers in each of the vectors.

Airfield Composition/Definition

Several variables are used to establish the composition and define the activities on the airfield. The three which require further definition are.

(1) distances between points on the airfield,
(2) rates of travel for aircraft and pilots, and
(3) number of shelters, revetments, and dispersal parking spots per squadron.

The distances between points on the airfield are entered in the matrix DIST (lines 574 to 587). DIST is a two-dimensional array. The distance between point I and J is entered in array location DIST(I,J) and DIST(J,I). The location codes for DIST are contained in lines 556 to 572.

Rates of travel are also inputs to the model. Rates are used to determine travel times from point I to point J on the airfield for both pilots and aircraft. These values are entered in lines 593 to 595.

The third variable, NTYPE (lines 500 to 508), is used to specify the number of each type of parking space per squadron. Since NPARK is dimensioned to 50, the sum of each type of parking for a squadron should not exceed 50. If a user specifies the number of shelters, dispersed spaces, and

revetments, for a squadron, and the total is less than 50, the coding will then assume the remaining spaces, up to 50, are dispersed spaces in the open. The numbers in lines 500 to 508 are the standard scenario numbers.

Setting the Master Clock

There are two areas to be addressed in controlling the flow of aircraft with the Executive Network. The first area is establishing the hours of daylight for the three days of the model. The second area is scheduling when the major events are to be initiated.

For the first area, this is accomplished by specifying the time of sunrise and sunset, or civil twilight (this is where the operational day is defined, i.e., the number of hours during which flying operations are conducted). These times are specified relative to TNOW, the simulation time. For example, on line 647 of Subroutine INTLC, daylight times are set to 15.0, 1455.0, and 2895.0 These values indicate to the model that it becomes daylight at 15 minutes into the simulation, and then again when TNOW equals 1455 and 2895 minutes. The DUSK variable is set in a similar fashion on line 648.

The other area, major event scheduling, is handled with a two- dimensional matrix called MAJEVNT (lines 653 to 700). The specifications of this variable dictate the time at which each major event is to be initiated relative to TNOW. Up to thirteen events can be scheduled. The first location of each pair event(1,1), is the time for the event to occur, and the second location of the pair, Event(1,2), is the event specified. Events which can be scheduled include.

0 - Terminate the run

1 - Start scheduling flights

2 - Begin night parking

3 - Perform QRA Changeover

4 - Determine replacement squadron requirements, and reconfigure aircraft for the following day's frag

99 - not used

The user must specify the times relative to TNOW, and the events to be initiated. As an example, lines 677 and 678 cause the night parking routine to begin parking aircraft in shelters when TNOW is equal to 980.0. Since this matrix, MAJEVNT, controls the model at the macro level, great care should be taken in setting these values.

Other Comments

Most, but not all of the variables covered in this section were of a more complex nature than those covered in the main text. Some were added because there were additional comments which needed to be made to a possible user who was interested enough to peruse these notes. These notes were designed to complement Chapter III and Appendices A and B. Anyone intending to use the model should also refer to Annex A.

To further aid in execution of the model a sample day file from a run on the Aeronautical Systems Division, CDC 6600 system (CYBER 74) is included in Figure C.3. This is not a small, quickly executed model. It requires an input-output time of around 400 seconds, and central processor times of around 140 seconds on the CYBER 74. Close attention should be given to setting the variables to the desired values on the first try.

```
1  CSA   NOS/BE L530C       L530C-CMR1 07/13/81
 21.21.30.RWM9NWC  FROM CSA/9N
 21.21.31.IP  00009600 WORDS - FILE INPUT  , DC 04
 21.21.31.RWM,CM250000,T400,IO400.  T000845,MANN,B
 21.21.31.0X4566,AFIT,AFIT,AFIT,91,91,91
 21.21.34.ATTACH,PROCFIL,ID=A810171,SN=ASDAD.
 21.21.34.PFN IS
 21.21.34.PROCFIL
 21.21.35.AT CY= 001 SN=ASDAD
 21.21.35.BEGIN,NOSFILE.
 21.21.35.RETURN,ZZZZZLB.
 21.21.35.ATTACH,ZZZZZLB,IFSLIB,PW=*---*,CY=999,ID
 21.21.35.=A810171,SN=ASDAD.
 21.21.36.LIBRARY,ZZZZZLB.
 21.21.37.NOTIFY.  NOSFILE VERSION 3 READY.
 21.21.38.         NOSFILE VERSION 3 READY.
 21.21.38.RETURN,PROCFIL.
 21.21.39.REVERT.
 21.21.39.GET,F708BIN,ID=COVEY.
 21.21.44.FILE NAME F708BIN    HAS BEEN RETRIEVED
 21.21.45.REWIND,F708BIN.
 21.21.45.ATTACH,PROCFIL,SLAMPROC,ID=AFIT.
 21.21.45.AT CY= 008 SN=AFIT
 21.21.45.BEGIN,SLAM,,M=F708BIN,PL=100000.
 21.21.46.IFE,NUM(0).EQ.0,NOPMD.
 21.21.47.ELSE,NOPMD.
 21.21.47.ATTACH,XXXSLAM,SLAM5,ID=AFIT,SN=AFIT.
 21.21.47.AT CY= 001 SN=AFIT
 21.21.47.IFE,NUM(F708BIN).EQ.1,MERGE.
 21.21.48.ELSE,MERGE.
 21.21.48.MAP,OFF.
 21.21.49.SEGLOAD,I=SLAMSEG.
 21.21.49.LOAD,F708BIN.
 21.21.49.LOAD,XXXSLAM.
 21.21.49.EXECUTE,,INPUT,OUTPUT,,,,*PL=100000.
```

Fig. C.3.1 Typical CYBER 74 Day File

```
  21.22.09.     NON-FATAL LOADER ERRORS -
21.22.09.DUPLICATE PROGRAM NAME FROM FILE
21.22.09.PROGRAM SKIPPED --- MAIN
21.22.09.LAST FILE ACCESSED- XXXSLAM
21.22.09.     NON-FATAL LOADER ERRORS -
21.22.09.DUPLICATE PROGRAM NAME FROM FILE
21.22.09.PROGRAM SKIPPED --- EVENT
21.22.09.LAST FILE ACCESSED- XXXSLAM
21.22.09.     NON-FATAL LOADER ERRORS -
21.22.09.DUPLICATE PROGRAM NAME FROM FILE
21.22.09.PROGRAM SKIPPED --- INTLC
21.22.09.LAST FILE ACCESSED- XXXSLAM
21.22.09.     NON-FATAL LOADER ERRORS -
21.22.09.DUPLICATE PROGRAM NAME FROM FILE
21.22.09.PROGRAM SKIPPED --- OTPUT
21.22.09.LAST FILE ACCESSED- XXXSLAM
21.22.09.     NON-FATAL LOADER ERRORS -
21.22.09.DUPLICATE PROGRAM NAME FROM FILE
21.22.09.PROGRAM SKIPPED --- USERF
21.22.09.LAST FILE ACCESSED- XXXSLAM
22.02.23.LOCKIN.
22.08.06.    STOP
22.08.06.    246700 MAXIMUM EXECUTION FL.
22.08.06.   128.774 CP SECONDS EXECUTION TIME.
22.08.07.ENDIF,MERGE.
22.08.08.ENDIF,NOPMD.
22.08.08.REVERT.CCL
22.08.09.OP  00007040 WORDS - FILE OUTPUT , DC 40
22.08.09.MS    7296  WORDS  (  124032 MAX USED)
22.08.09.CPA    131.397 SEC.      107.078 ADJ.
22.08.09.IO     324.551 SEC.       96.067 ADJ.
22.08.09.CM   36109.804 KWS.      170.098 ADJ.
22.08.09.CRUS                     373.244
22.08.09.COST                $     24.60
22.08.09.PP     231.949 SEC.     DATE 01/27/82
22.08.09.EJ  END OF JOB, 9N  T000845.
```

Fig. C.3.2 Typical CYBER 74 Day File

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER AFIT/GST/OS/82M-9	2. GOVT ACCESSION NO. AD-A115 689	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) A USER DEFINABLE SLAM AIRFIELD MODEL DESIGNED FOR EXPERIMENTATION AND ANALYSIS (Volume II)		5. TYPE OF REPORT & PERIOD COVERED Master's Thesis 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Robert W. Mann, Maj, USAF Brian J. Shook, 2Lt, USAF		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Air Force Institute of Technology (AFIT/EN) Wright-Patterson AFB, OH 45433		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Air Force Institute of Technology (AFIT/EN) Wright-Patterson AFB, OH 45433		12. REPORT DATE March 1982 13. NUMBER OF PAGES 250
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) UNCLASSIFIED 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

4 JUN 1982

APPROVED FOR PUBLIC RELEASE: IAW AFR 190-17

LYNN E. WOLAVER
Dean for Research and Professional Development

AIR FORCE INSTITUTE OF TECHNOLOGY (ATC)
WRIGHT-PATTERSON AFB, OH 45433

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Air Force Facilities	Airfield
Airports	Airbase
Computerized Simulation	Model
Targets	

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This research effort was undertaken to investigate a methodology for determining the most critical elements on a fighter-bomber airbase with respect to sorties generated over a three-day period. The methodology is founded on a user definable computer simulation model written in SLAM (FORTRAN based) and supported by several FORTRAN routines. The remainder of the

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

Block 20:

methodology entails the use of factorial experimental designs for examining airfield element criticality. The airfield elements are the experimental factors. They are set to user specified levels according to the experimental design. The model produces a single response variable--sorties generated over a three-day period. Results are analyzed with common statistical techniques (Method of Contrasts, ANOVA, Duncan's Multiple Range Test). Special attention was placed on documentation of the model to insure ease of implementation by a user. Model usage is demonstrated with two experiments and their analysis. Because this methodology does not require Monte Carlo simulation of damage to the airfield, the determination of element criticality is straightforward. The lucrative targets on the airfield are then the most critical elements which can be effectively attacked with available weapons and delivery systems.